

Evaluation of the DRAGON Code for VHTR Design Analysis

prepared by
Nuclear Engineering Division
Argonne National Laboratory

About Argonne National Laboratory

Argonne is managed by The University of Chicago for the U.S. Department of Energy under contract W-31-109-Eng-38. The Laboratory's main facility is outside Chicago, at 9700 South Cass Avenue, Argonne, Illinois 60439. For information about Argonne and its pioneering science and technology programs, see www.anl.gov.

Availability of This Report

This report is available, at no cost, at <http://www.osti.gov/bridge>. It is also available on paper to U.S. Department of Energy and its contractors, for a processing fee, from:

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831-0062
phone (865) 576-8401
fax (865) 576-5728
reports@adonis.osti.gov

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor The University of Chicago, nor any of their employees or officers, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of document authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, Argonne National Laboratory, or The University of Chicago.

Evaluation of the DRAGON Code for VHTR Design Analysis

by
T.A. Taiwo and T.K. Kim
Nuclear Engineering Division, Argonne National Laboratory

September 30, 2005



THE UNIVERSITY OF
CHICAGO

Argonne National Laboratory is managed by
The University of Chicago for the U.S. Department of Energy

Table of Contents

	Page
Abstract	1
1.0 INTRODUCTION	3
2.0 EVALUATION OF DRAGON CODE FOR GRAPHITE-MODERATED REACTORS	5
2.1 DRAGON VHTR Deterministic Lattice Models.....	5
2.2 Assessment of DRAGON for VHTR and GT-MHR Burner Fuel Elements.....	5
2.3 Evaluation of the DRAGON Code for Liquid-Salt-Cooled Very High Temperature Reactor (LS-VHTR) Analysis	9
3.0 EVALUATION OF DRAGON LIBRARIES.....	12
3.1 General Descriptions of WIMS-D Format Libraries	12
3.2 Impact of Q-Values.....	18
3.3 Performance of WIMS-D Libraries	21
4.0 DRAGON CODE WORKSHOP	24
5.0 CONCLUSIONS AND RECOMMENDATIONS	25
ACKNOWLEDGMENTS	26
REFERENCES	27
APPENDIX A.....	29

List of Tables

	Page
Table 1. Comparison of Deterministic and Monte Carlo Eigenvalue Results for NGNP and GT/AD-MHR Assembly Problems	6
Table 2. Magnitude of Assembly Double Heterogeneity Effect ($\% \Delta \rho$)	7
Table 3. Comparison of k_{inf} Values for Fuel Element at Cold State	10
Table 4. Double Heterogeneity Effect ($\Delta \rho$) for LS-VHTR	10
Table 5. Comparison of Void Reactivity Coefficients (pcm/%void).....	11
Table 6. List of WIMS-D Format Libraries	13
Table 7. Explicitly Represented Fission Products and Fission Yields.....	15
Table 8. Comparison of Q-Values (MeV per Fission).....	16
Table 9. Comparison of Eigenvalues for Infinite Cell Benchmarks	22
Table 10. VHTR Assembly Eigenvalues with Different Libraries	23

List of Figures

	Page
Figure 1. GT/AD-MHR TRU Spectrum calculated by MCNP4C, WIMS8 and DRAGON.....	8
Figure 2. VHTR/NGNP Assembly Spectrum calculated by MCNP4C, WIMS8 and DRAGON.....	8
Figure 3. Actinide Chains of IAEA WLUP Libraries	17
Figure 4. Comparison of Eigenvalues from Different Implementations of Depletion Chains.....	19
Figure 5. Depletion Results with Original Q-Values of Libraries	20
Figure 6. Depletion Results with Modified WIMS-ANL Q-Values	21

Abstract

This letter report summarizes three activities that were undertaken in FY 2005 to gather information on the DRAGON code and to perform limited evaluations of the code performance when used in the analysis of the Very High Temperature Reactor (VHTR) designs. These activities include:

- Use of the code to model the fuel elements of the helium-cooled and liquid-salt-cooled VHTR designs. Results were compared to those from another deterministic lattice code (WIMS8) and a Monte Carlo code (MCNP).*
- The preliminary assessment of the nuclear data library currently used with the code and libraries that have been provided by the IAEA WIMS-D4 Library Update Project (WLUP).*
- DRAGON workshop held to discuss the code capabilities for modeling the VHTR.*

1. INTRODUCTION

A previous assessment of deterministic lattice physics codes revealed there are only a few codes that allow the detailed treatment of the double heterogeneity effect inherent in the Very High Temperature Reactor (VHTR) design at the fuel element (assembly) level. [1] Two such codes, DRAGON and WIMS8, [2,3] are available for use at ANL. The WIMS8 code is a proprietary code that has a yearly fee. The access to the WIMS8 source code is also quite expensive. On the other hand, the DRAGON code is quite attractive to the Gen IV program because it is open software that could be additionally developed to meet the specific needs of the program with only limited incremental cost. The code was developed by researchers at the Ecole Polytechnique de Montreal, Canada and is currently maintained by this institute.

The DRAGON code has a collection of models for simulating the neutronic behavior of a unit cell or a fuel lattice in a nuclear reactor. [2] The typical functionalities found in most modern lattice codes are contained in DRAGON. These include interpolation of microscopic cross sections supplied by means of standard libraries; resonance self-shielding calculations in multidimensional geometries; multigroup and multidimensional neutron flux calculations which can take into account neutron leakage; transport-transport or transport-diffusion equivalence calculations; and modules for editing condensed and homogenized nuclear properties for reactor calculations.

The current version of the code contains three algorithms for the solution of the integral transport equation, ranging from a simple collision probability method coupled with the interface current method to the full collision probability method. The code also performs isotopic depletion calculations. The code user must however supply cross sections in one of the following standard formats: DRAGLIB, MATXS (TRANSX-CTR), WIMSD4, and WIMS-AECL. Macroscopic cross sections can also be read by DRAGON via the input data stream. At ANL, the 69- and 172-group cross section libraries created in WIMSD4-format by the Reduced Enrichment for Research and Test Reactors (RERTR) project are used with the DRAGON code. The depletion chains and types of fission products to be tracked by the code are obtained from the cross section library used. For the DRAGON calculations, cross section data for the heavy

nuclides are tabulated at different temperatures (2-4 points) and all the heavy nuclides contained in the library are treated as resonance materials.

An attractive feature of the DRAGON code is its ability to treat particulate fuel in a graphite matrix in a full-assembly calculation. This capability has been used for modeling the fuel assemblies of block-type, high-temperature gas-cooled thermal reactors and the pebble elements in alternative pebble-bed concepts.

An evaluation of the DRAGON lattice capability for use in the design of the Very High Temperature Reactor (VHTR) system was initiated this year (FY 2005) under the Gen IV program. The activities performed this year can be classified into the following three areas:

- DRAGON code was used to model the fuel elements of the helium-cooled and liquid-salt-cooled VHTR designs. Results were compared to those from another deterministic lattice code (WIMS8) and a Monte Carlo code (MCNP).
- The nuclear data library currently used with the code was evaluated by comparing the code performance against those obtained when libraries made available by the IAEA WIMS-D4 Library Update Project (WLUP) are used. This step was also an evaluation of the WLUP libraries to determine their potential for use in future DRAGON calculations.
- A DRAGON workshop was held to discuss the code capabilities for modeling the VHTR.

These three activities are summarized in Sections 2, 3, and 4, respectively.

2. EVALUATION OF DRAGON CODE FOR GRAPHITE-MODERATED REACTORS

2.1 DRAGON VHTR Deterministic Lattice Models

The DRAGON code does not have an explicit model for representing the prismatic-block VHTR fuel element. In the DRAGON full-assembly model that has been developed at Argonne, the fuel element is formed by a collection of pin-cell sized hexagons. Each pin-cell contains the fuel compact and its surrounding block graphite. When all the fuel and coolant-hole pin-cells are represented, the block graphite content is not totally accounted for and therefore an extra ring of pin-cell sized hexagons is used to represent the remaining graphite. The number density of the graphite in these peripheral cells is modified to preserve the graphite content of the assembly block. Due to the use of the pin-cell sized hexagons, the DRAGON assembly model has jagged boundaries, not the flat boundaries of the hexagonal block.

2.2 Assessment of DRAGON for VHTR and GT-MHR Burner Fuel Elements

The DRAGON code has been used for the modeling of the VHTR/Next Generation Nuclear Plant (NGNP). [4] In that study, two kinds of prismatic fuel elements were considered: fuel elements of the NGNP and GT-MHR Minor Actinide Burner (GT/AD-MHR) designs. Both design concepts are alternative designs of the General Atomics (GA) Gas Turbine-Modular Helios Reactor (GT-MHR). The NGNP is being developed under the Gen-IV program to provide very high temperature gas for both electricity and hydrogen production. The GT/AD-MHR was proposed by GA for deep burning of the minor actinides. The assembly dimensions, materials, and composition data for these designs can be found in Reference 4.

The performance of the DRAGON and WIMS8 codes was evaluated by comparing results from the codes to those from the high fidelity MCNP4C code. The MCNP4C calculations for the fuel element were performed using the ENDF/B-VI based nuclear data library distributed with the code. The calculations were for the cold state (294 K). A million neutron histories are utilized in each MCNP4C calculation, with the first 50,000 histories ignored to allow convergence of the fission source before averaging the k_{∞} .

The results for fuel element problems are summarized in Tables 1 and 2. Both NGNP ($\text{UC}_{0.5}\text{O}_{1.5}$ fuel) and GT/AD-MHR results (with $\text{TRUO}_{1.7}$ and UO_2 fuels) are reported in the tables. Note that the difference between cases is defined using $\Delta\rho = \Delta k/k_1k_2$. The results in Table 1 show that eigenvalues predicted with the deterministic DRAGON and WIMS8 codes generally agree well with the reference values obtained from MCNP4C calculations with stochastic distribution models. However, WIMS8 consistently underestimates the assembly eigenvalue for all the cases considered, while DRAGON overestimates it. For the GT/AD-MHR UO_2 assembly, DRAGON and WIMS8 show excellent agreements with MCNP4C. The DRAGON and WIMS8 errors are increased in the GT/AD-MHR TRU and VHTR/NGNP fuel element problems that have much larger resonance absorption. This suggests a potential need to improve the resonance absorption treatments in DRAGON and WIMS8.

Table 1. Comparison of Deterministic and Monte Carlo Eigenvalue Results for NGNP and GT/AD-MHR Assembly Problems.

Computation model	NGNP, $\text{UC}_{0.5}\text{O}_{1.5}$	GT-MHR, UO_2	GT-MHR, $\text{TRUO}_{1.7}$
Stochastic Average	1.53280 ± 0.00082	1.57335 ± 0.00040	1.25838 ± 0.00040
DRAGON	1.54393 (470)*	1.57565 (93)	1.26794 (599)
WIMS8	1.52993 (-122)	1.57121 (-86)	1.25326 (-325)

* Difference between deterministic (DRAGON or WIMS8) and stochastic (MCNP4C) average eigenvalues in pcm.

The double heterogeneity effect for the fuel assemblies have been calculated using the MCNP4C, WIMS8, and DRAGON codes. The double heterogeneity effect measures the difference in the assembly eigenvalue calculations performed with detailed heterogeneous compact model (coated fuel particle layers and compact graphite matrix are explicitly represented) and a homogenized model in which the various material components of the compact have been volume weighted to give a single material composition. For consistent comparison of WIMS8 calculations, the calculation with homogeneous compact model was performed using a two-step procedure employed for the heterogeneous compact model; [1] the super-cell

calculation was done using the PIJ module and the subsequent assembly calculation using the CACTUS module. It is noted that the CACTUS module has an option for treating directly such a singly homogeneous-compact lattice problem.

Results for the pin-cell and assembly calculations showed that the double heterogeneity effect in an assembly problem is slightly smaller than that of a corresponding unit pin-cell problem, but it is still significant. The reduced resonance absorption resulting from additional moderation provided by the graphite outside fuel cells decrease the double heterogeneity effect in assembly problems. The double heterogeneity effects for the assembly are summarized in Table 2. Compared to neutron capture in U-238 of the GT/AD-MHR UO₂ assembly, the capture in U-238 of the NGNP assembly is 2.6 times larger due to a higher packing fraction of fuel particles and a bigger size of fuel kernel. (The amount of U-238 in the NGNP assembly is six times larger than that of GT/AD-MHR UO₂ assembly.) As a result, the double heterogeneity effect is about two times larger in the NGNP assembly. The double heterogeneity effect is quite large in the TRU fuel assembly because of the large amount of Pu-239 and Pu-240.

Table 2. Magnitude of Assembly Double Heterogeneity Effect (% $\Delta\rho$).

Computation Model	NGNP, UC _{0.5} O _{1.5}	GT-MHR, UO ₂	GT-MHR, TRUO _{1.7}
MCNP4C	2.3	1.4	13.1
DRAGON	2.9	1.5	13.1
WIMS8	2.7	1.4	12.9

Comparison of the fuel element flux spectra calculated by the codes has also been done and results for two cases are summarized in Figures 1 and 2, for the cases using the detailed heterogeneity model. Generally, the spectra calculated by the deterministic codes are very similar to the results of the MCNP4C calculations. A slight over-estimation of the thermal neutron flux peak is observed for the DRAGON results. The reason for this has not been determined and would require additional study.

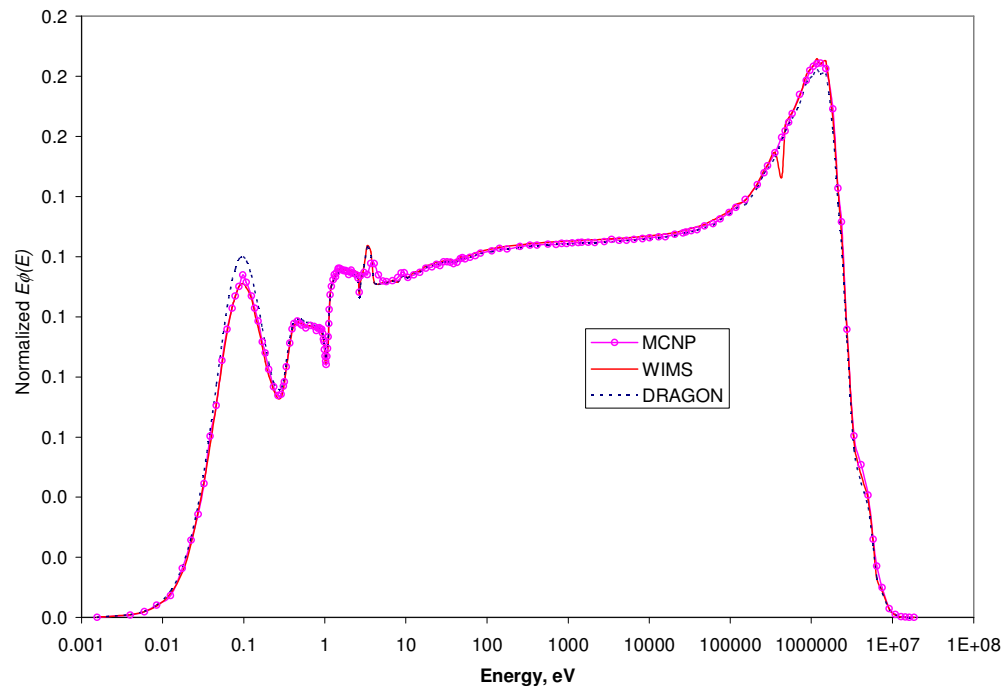


Figure 1. GT/AD-MHR TRU Spectrum calculated by MCNP4C, DRAGON and WIMS8.

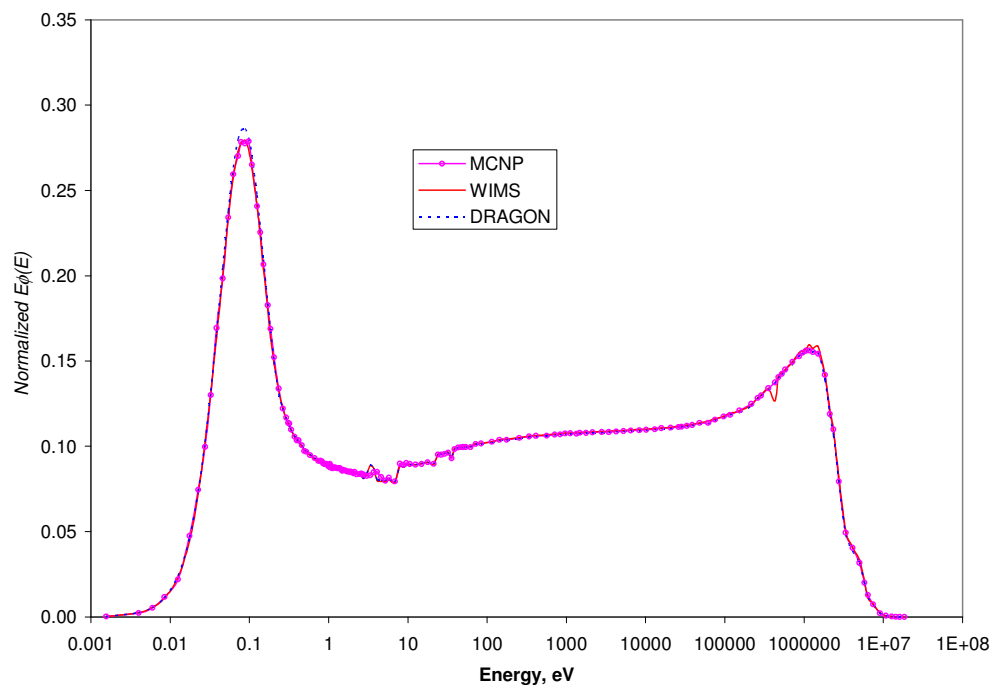


Figure 2. VHTR/NGNP Assembly Spectrum calculated by MCNP4C, DRAGON and WIMS8.

2.3 Evaluation of the DRAGON Code for Liquid-Salt-Cooled Very High Temperature Reactor (LS-VHTR) Analysis

The study presented in Section 2.2 was performed to evaluate the performance of the DRAGON and WIMS8 code models for the analysis of the helium-cooled VHTR. The study demonstrated that the codes are capable of calculating lattice parameters of interest very accurately compared to Monte Carlo reference solutions in the range of interest. While the helium-cooled VHTR and the liquid-salt cooled VHTR employ the same assembly design, the different characteristics of the coolants suggest the need for further verification of the performance of the lattice codes.

The performance of the deterministic lattice codes DRAGON and WIMS8 for analyzing the LS-VHTR fuel element has been evaluated by comparing results from the codes to those from the Monte Carlo code MCNP4C. [5] The MCNP4C calculations for the fuel assembly were performed using the ENDF/B-VI nuclear data library distributed with the code. In the DRAGON and WIMS8 calculations, the 172 group transport equations were solved. Information on the fuel element design can be found in Reference 5. In the study, the packing fraction, uranium enrichment and coolant-hole diameter are 25%, 10% and 0.953 cm respectively. A lithium enrichment value of 99.995% (Li-7 content) is used. [5] Results from the study are summarized in Tables 3 and 4. In addition to the traditional element eigenvalue, an estimate of the double heterogeneity effect was also obtained. This effect was determined by taking the difference in the multiplication factor (k_{inf}) values derived from calculations using the spatially *heterogeneous* and *homogeneous* compact models. In the heterogeneous model the coated fuel particles are explicitly represented in the compact. On the other hand, in the homogeneous model, the compositions of the coated fuel particles are smeared with those of the graphite matrix using volume weighting. In the latter model, the self-shielding effect of the fuel at the particle level is not represented.

Table 3 summarizes the eigenvalues calculated by the DRAGON and WIMS8 lattice codes and the MCNP4C Monte-Carlo code. The DRAGON and WIMS8 codes calculated fuel element k_{inf} values that are within 100 pcm and 200 pcm, respectively, of those from MCNP4C calculations. (The reactivity differences were computed using $\Delta\rho = \Delta k/k_1 k_2$.) Components of

these differences come from the different nuclear data files used in the calculations (e.g., JEF2.2 for WIMS8). These differences are similar to those observed in Section 2.2.

Table 3. Comparison of k_{inf} Values for Fuel Element at Cold State.

Compact model	Li Enrichment	Code	Eigenvalue	Difference from reference, pcm $\Delta\rho$
Heterogeneous	99.995 %	MCNP4C	1.53861 ± 0.00067	Reference
		WIMS8	1.53756	-44
		DRAGON	1.53832	-12
Homogeneous	99.995 %	MCNP4C	1.47886 ± 0.00075	Reference
		WIMS8	1.47681	-94
		DRAGON	1.48216	151

The deterministic and Monte Carlo codes predicted very similar values for the double heterogeneity effect in the LS-VHTR fuel element (difference in k_{inf} from calculations using the smeared and explicit heterogeneous fuel compact models). The MCNP4C calculation predicted a value of 2.6% $\Delta\rho$ (see Table 4) and the two lattice codes give a deviation of about 0.1% $\Delta\rho$ for these cases; these are surprisingly small differences. For comparison, an earlier MCNP4C calculation for the helium-cooled VHTR fuel element gave a value of 2.3% $\Delta\rho$ for the double heterogeneous effect.

Table 4. Double Heterogeneity Effect ($\Delta\rho$) for LS-VHTR.

Code	Lithium Enrichment (%Li-7 Content)	Double Heterogeneity Effect, % $\Delta\rho$
MCNP4C	99.995	2.6
WIMS8	99.995	2.7
DRAGON	99.995	2.5

Also in the same LS-VHTR study, [5] the coolant void reactivity coefficient (CVRC) was estimated as a function of burnup for the reference fuel-element design, using the deterministic lattice codes DRAGON and WIMS8. The burnup-dependent CVRC were obtained by branch calculations at the burnup points of interest using full voiding of the coolant to derive the 100% voiding reactivity effect. The result is then normalized to the reactivity change per percent voiding; Monte Carlo calculations were not available for these state. For the fuel element CVRC study, an assembly fuel enrichment of 15% and fuel packing fraction of 25% were used.

Results of the study are summarized in Table 5. Noticeable differences are observed in the CVRC values calculated by the DRAGON and WIMS8 codes. The differences between the two codes were also found to become larger with burnup. The reason for these differences has not been completely investigated in the current study. It is however probably due to differences in basic cross section data (JEF2.2 for WIMS8 and ENDF/B-VI for DRAGON) and differences in material compositions at burnup points arising from different solution methodology and code data. The material composition differences with burnup have been attributed to the differences in the energy released per fission values used in the two codes. This effect is investigated in Section 3.2.

It is noted that the magnitude of the differences in the CVRC is actually small, the largest difference being about 600 pcm over 100% voiding for the case with lithium enrichment of 99.99% and about 300 pcm over 100% voiding for the case with lithium enrichment of 99.995%. The results also show that the magnitude of the positive CVRC is small, being less than \$1 per 100% voiding (assuming a delayed neutron fraction value of 600 to 700 pcm). Further evaluations of the differences with burnup, however, need to be performed.

Table 5. Comparison of Void Reactivity Coefficients (pcm/% void).

	Case with Li enrichment of 99.99%		Case with Li enrichment of 99.995%	
Burnup (GWd/t)	WIMS8	DRAGON	WIMS8	DRAGON
0.0	3.6	5.4	0.2	1.5
15.0	3.0	2.9	-0.4	-0.2
30.0	1.6	0.4	-1.8	-2.4
45.0	0.6	-2.0	-3.0	-4.2
60.0	-0.2	-4.2	-4.0	-6.1
90.0	-1.0	-7.1	-5.3	-8.6

Additionally, the results indicate that the DRAGON and WIMS8 codes give similar trends for the CVRC. It is observed that the coefficient is positive at the beginning of cycle and decreases with increasing burnup. At a burnup between 15 and 20 GWd/t, the fuel element CVRC becomes negative in the case using a lithium enrichment of 99.995%.

3. EVALUATION OF DRAGON LIBRARIES

The DRAGON code can read several standard library formats such as DRAGLIB, MATXS, WIMS-D, and WIMS-AECL. Macroscopic cross sections can also be read via the input data stream. [2] The 69- and 172-group cross section libraries used with the DRAGON code at ANL have been based on the WIMS-D4 format. These ANL libraries will be referred to as WIMS-ANL in this section. Previous clean comparisons of the DRAGON code results (using the WIMS-ANL library) to other lattice and Monte Carlo codes have shown that DRAGON has promise as a lattice capability for analyzing VHTRs (see for example Sections 2.2 and 2.3). However, more detailed evaluations have revealed the need for additional assessment of the neutron cross sections that are typically used with it at ANL (no data library is distributed with the DRAGON code); see the comparison of the CVRC variation with burnup in Section 2.3.

The WIMS-ANL library is based on the WIMSD4 cross-section data-library format that is one of the popular ones used in several lattice codes (DRAGON, WIMS-D4, etc). This library needs to be updated as new revisions to the base nuclear data file (ENDF/B in the U.S. case) are released. The validation of the WIMS-ANL file requires great effort and alternative approaches are being investigated to minimizing this effort at ANL. At the same time, WIMS-format libraries based on recent evaluated nuclear data files have been created for general use under the IAEA WIMS-D Library Update Project (WLUP) since 1990. Several libraries based on this format are currently open to the public through an IAEA website (<http://www-nds.iaea.org/wimsd/downloads.htm>). It was decided at ANL to obtain these WLUP files, exercise (assess) them to determine their pedigree and then make a determination of their applicability to reactor physics calculations for the VHTR design. A study was therefore conducted to evaluate the performance of the WIMS libraries available from ANL and the IAEA WLUP by intercomparison of DRAGON code results, and for some cases comparison to MCNP4C (Monte Carlo) results. The results of that study are summarized in this section.

3.1 General Descriptions of WIMS-D Format Libraries

The IAEA supported WIMS-D Library Update Project (WLUP) had the goal of providing libraries compatible with the WIMS-D family of codes or equivalent lattice codes. This would

enable the use of the most recent evaluated nuclear data files (ENDF/B-VI.8, JEF2.2, JENDL3.2, etc) for both thermal research and power reactor calculations. Table 6 summarizes the WIMS-D format libraries available for use in the current study and sources of the evaluated nuclear data files. In this work, most of the DRAGON calculations were performed using the 69- and 172-group WIMS-ANL and IAEA recommended libraries.

Table 6. List of WIMS-D Format Libraries.

Library name	Groups	Description
WIMS-ANL	69	ANL developed based on ENDF/B-VI data
	172	
IAEA	69	Recommended by IAEA-WLUP participants
	172	
IAEA-ENDFB6	69	Library based on ENDF/B-VI evaluated nuclear data
	172	
IAEA-JEF2.2	69	Library based on JEF2.2 evaluated nuclear data
	172	
IAEA-JENDL3	69	Library based on JENDL3.2 evaluated nuclear data
	172	

Some similarities and differences exist between the nuclear-data library files. Both the WIMS-ANL and IAEA recommended libraries have been created using the WIMSR module of the NJOY code. Generally, most of the cross sections types (transport, absorption, P_1 transport-corrected scattering matrix, resonance integrals, potential cross sections, etc) included in the libraries are the same. However, the total number of fission products, energy released per fission values, and actinide chains (in particular, Am-241, U-233, U-238, etc) are different.

The resonance parameters of the WIMS-D format library are created assuming that the intermediate resonance approximation and equivalence theorem are used in the lattice code employing the library. In the library, the resonance integrals are tabulated as functions of temperature and background cross section (σ_b). The intermediate parameters (i.e., Goldstein-Cohen λ factor) are stored for each resonance absorber and resonance group. [6]

Since some of the fission products have a significant impact on the nuclear performance of the core, they must be represented explicitly. Although individually unimportant, others have a cumulative effect that cannot be neglected; usually they are lumped together and called pseudo

fission product. In the IAEA WLUP libraries, 56 nuclides are explicitly represented and 79 nuclides are lumped into a pseudo fission product. The criteria for selecting the fission products that are represented explicitly in the WLUP are provided in Reference 6. Thirty-four fission products are explicitly represented in the WIMS-ANL library. Table 7 compares the fission yields for the primary fissionable nuclides (U-235 and Pu-239) contained in the IAEA WLUP and WIMS-ANL libraries. Values from the WIMS8 library are also included for comparison. In this table, the “IAEA” library denotes the WIMS-D format library recommended by the WLUP participants. Note that the cross sections for nuclides contained in the WIMS8 library were created using JEF2.2 nuclear data. [6] Although about 100 fission products are explicitly represented in the WIMS8 library, only the nuclides appearing in the IAEA library are compared in Table 7.

In the ANL library, many more fission products are ignored compared to the IAEA libraries because the fission yield values for these nuclides are negligibly small and their reactivity impacts have been found to be small in core calculations. Except for the lumped fission product, the fission yield values of the IAEA and WIMS-ANL libraries are quite similar because both fission yield values have been taken from the ENDF/B-VI nuclear data file. However, the fission yield values of the WIMS8 library are slightly different from those of the other two because of the different sources.

The effective energy released per fission is required for burnup calculations. In the IAEA WLUP WIMS-D format libraries, the effective energy released per fission (simply Q-value) includes the contributions from fission, from the kinetic energy of incident neutrons, and from the decay of the capture products:

$$Q^i = Q_f^i + Q_c^i + Q_n^i,$$

where Q_f is the energy released in fission excluding the energy carried away by neutrinos and the kinetic energy of incident neutrons, Q_c is the contribution from the decay of the capture products, and Q_n is the kinetic energy of the incident neutrons. Index i refers to a fissionable nuclide in all cases. Among the three contributions, the Q_f is dominant.

Table 7. Explicitly Represented Fission Products and Fission Yields.

	WIMS-D Format Library				WIMS8 library	
	IAEA		ANL			
	U235	Pu239	U235	Pu239	U235	Pu239
Krypton-83	0.005	0.003	0.005	0.003	0.005	0.003
Molybdenum-95	0.065	0.048	0.065	0.048	0.000	0.000
Technetium-99	0.061	0.062	0.061	0.062	0.062	0.062
Ruthenium-101	0.052	0.060	0.052	0.060	0.051	0.062
Ruthenium-103	0.030	0.070	0.030	0.070	0.030	0.070
Ruthenium-106	0.004	0.044	N/A	N/A	N/A	N/A
Rhodium-103	0.000	0.000	0.000	0.000	0.000	0.000
Rhodium-105	0.010	0.056	0.010	0.056	0.000	0.000
Palladium-105	0.000	0.000	0.000	0.000	0.000	0.000
Palladium-107	0.001	0.033	N/A	N/A	0.001	0.032
Palladium-108	0.001	0.022	0.001	0.020	0.001	0.021
Silver-109	0.000	0.015	0.000	0.010	0.000	0.015
Cadmium-113	0.000	0.001	0.000	0.001	0.000	0.001
Indium-115	0.000	0.000	0.000	0.001	0.000	0.000
Antimony-125	0.000	0.001	N/A	N/A	N/A	N/A
Tellurium-127m	0.000	0.001	N/A	N/A	N/A	N/A
Iodine-127	0.001	0.004	0.002	0.005	0.001	0.005
Xenon-131	0.063	0.065	0.063	0.061	0.063	0.062
Cesium-133	0.029	0.039	0.029	0.039	0.000	0.000
Cesium-134	0.078	0.077	N/A	N/A	0.077	0.076
Cesium-137	0.003	0.011	0.003	0.015	0.003	0.010
Iodine-135	0.061	0.069	N/A	N/A	0.063	0.069
Xenon-134	0.067	0.070	0.067	0.070	0.000	0.000
Xenon-135	0.000	0.000	0.000	0.000	0.000	0.000
Cesium-135	0.000	0.000	0.000	0.000	0.000	0.000
Xenon-136	0.063	0.067	N/A	N/A	0.062	0.065
Neodymium-143	0.060	0.044	0.060	0.043	0.000	0.000
Neodymium-145	0.039	0.030	0.039	0.030	0.039	0.031
Promethium-147	0.011	0.009	0.011	0.009	0.000	0.000
Promethium-147*	0.012	0.011	0.012	0.011	0.000	0.000
Promethium-148	0.000	0.000	0.000	0.000	0.000	0.000
Promethium-148 ^m	0.000	0.000	0.000	0.000	0.000	0.000
Promethium-149	0.011	0.012	0.011	0.012	0.010	0.013
Samarium-147	0.000	0.000	0.000	0.000	0.000	0.000
Samarium-148	0.000	0.000	N/A	N/A	0.000	0.000
Samarium-149	0.000	0.000	0.000	0.000	0.000	0.000
Samarium-150	0.000	0.000	0.000	0.000	0.000	0.000
Samarium-151	0.004	0.007	0.004	0.008	0.000	0.000
Samarium-152	0.003	0.006	0.003	0.006	0.003	0.006
Europium-151	0.000	0.000	N/A	N/A	N/A	N/A
Europium-152	0.000	0.000	N/A	N/A	N/A	N/A
Europium-153	0.002	0.004	0.002	0.004	0.000	0.000
Europium-154	0.000	0.000	0.000	0.000	0.000	0.000
Europium-155	0.000	0.002	0.000	0.002	0.000	0.002
Gadolinium-154	0.000	0.000	N/A	N/A	N/A	N/A
Gadolinium-155	0.000	0.000	N/A	N/A	0.000	0.000
Gadolinium-156	0.000	0.001	N/A	N/A	0.000	0.000
Gadolinium-157	0.000	0.001	0.000	0.001	0.000	0.000
Gadolinium-158	0.000	0.000	N/A	N/A	0.000	0.000
Dysprosium-160	0.000	0.000	N/A	N/A	N/A	N/A
Dysprosium-161	0.000	0.000	N/A	N/A	0.000	0.000
Dysprosium-162	0.000	0.000	N/A	N/A	0.000	0.000
Dysprosium-163	0.000	0.000	N/A	N/A	0.000	0.000
Dysprosium-164	0.000	0.000	N/A	N/A	N/A	N/A
Holmium-165	0.000	0.000	N/A	N/A	N/A	N/A
Erbium-166	0.000	0.000	N/A	N/A	N/A	N/A
Erbium-167	0.000	0.000	N/A	N/A	N/A	N/A
Lumped FP	1.069	0.867	0.304	0.329	2.000	2.000

*Decays to meta-stable Promethium-148

In Table 8, the Q-values of the different libraries are compared. For the comparison, the Q-values used in the WIMS8 library are included in this table. The Q-values of the IAEA and WIMS8 libraries are quite similar, while those of the WIMS-ANL library are consistently smaller than the others. This indicates that the Q-values of the WIMS-ANL library do not include the minor contributions (such as capture or kinetic energy of incident neutrons, or both). The impact of the different Q-values will be discussed in Section 3.2.

Table 8. Comparison of Q-Values (MeV per Fission).

Library	ANL WIMS-D	IAEA WIMS-D	WIMS8
Th232	188.5	192.8	192.8
Pa233	189.8	196.6	196.6
U233	191.0	199.1	199.1
U234	190.3	199.2	199.2
U235	193.7	201.7	201.7
U236	194.5	201.8	201.8
U237	200.0	202.0	202.0
U238	198.1	205.0	205.0
Np237	196.4	205.1	205.1
Np239	190.0	208.0	208.0
Pu238	197.4	208.5	208.5
Pu239	199.9	210.0	210.0
Pu240	199.5	210.4	210.4
Pu241	202.0	212.9	212.9
Pu242	201.6	213.1	213.1
Am241	202.3	214.7	214.7
Am242m	202.3	215.8	215.8
Am243	203.6	217.0	217.0
Cm242	202.5	221.4	221.4
Cm243	204.0	220.3	220.3
Cm244	203.0	220.7	220.7
Cm245	200.0		221.1

Figure 3 illustrates the actinide chains contained in the IAEA WIMS-D format libraries. The WIMS-D format does not contain the $(n,2n)$ or $(n,3n)$ cross sections, but the total reaction rate is preserved by correcting the absorption cross section. However, the neglect of these reactions might result in the underestimation of some nuclide densities (for examples, U-237, Pu-238, etc). To compensate the underestimation, two pseudo nuclides are introduced in the U-233 and U-238 chains in Figure 3. In WIMS-D format libraries, these pseudo nuclides are treated as

fission products of U-233 and U-238 and decay instantly to U-232 and U-237, respectively. The fission yield values for these pseudo nuclides are obtained from the ratio of $(n,2n)/(n,f)$ reactions.

There are two states of Am-242: stable (Am-242g) and meta-stable (Am-242m). The ground state Am-242 decays to Cm-242 (beta decay with 16.02 hours half-life) and Pu-242 (electron capture with 4.2 ms half-life). The branching ratio to Am-242m is dependent on the neutron spectrum, but it is about 14% in a typical thermal spectrum. [3] In the IAEA libraries, a simplification is introduced to represent the Am-241 and higher actinide chains because only one nuclide can be specified as the capture product in the WIMS-D format.

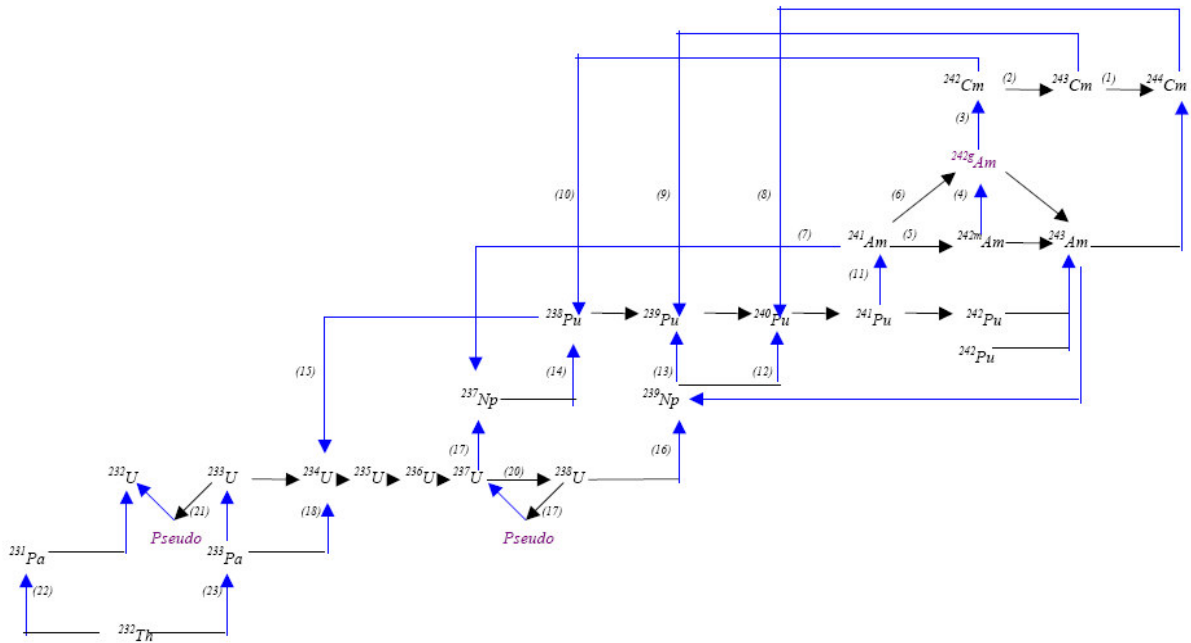


Figure 3. Actinide Chains of IAEA WLUP Libraries. [6]

The meta-stable isotope (Am-242m) in the IAEA libraries is specified as a capture product explicitly with a branching ratio B_{Am2m} , while the ground state isotope (^{242g}Am in Figure 3) is treated as a fission product with a fission yield value of $(c/f) \times (1 - B_{Am2m})$. Here, c/f denotes the capture-to-fission ratio of Am-241. In Figure 3, ^{242g}Am decays to Cm-242, or goes to Am-243 after neutron capture, and the electron capture (i.e., decay to Pu-242) of Am-242g is ignored.

Similar simplification is observed in the ANL library representation of the Am-241 and higher actinide chains. The meta-stable isotope (Am-242m) is specified as the capture product with a branching ratio B_{Am2m} , but the ground state isotope is skipped and Am-241 decays to Cm-242 directly with $(1 - B_{Am2m})$ branching ratio. Thus, ANL library ignores the production of Pu-242 from the electron capture of Am-242.

In the WIMS8 actinide chains, Am-242 is bypassed because its half-life is very short and Am-241 goes to Am-242m, Cm-242, and Pu-242 directly with appropriate branching ratios. For a typical thermal system, these branching ratios are 14%, 71%, and 15%, respectively. This indicates that about 15% of Pu-242 production from the Am-241 is ignored in both WIMS-D format libraries.

3.2 Impact of Q-Values

The comparison of the DRAGON and WIMS8 calculated CVRC variation with burnup in Section 2.3 shows the increasing deviation of the code results for this parameter with burnup. This trend has previously been observed for other parameters (even for the fuel element eigenvalue, k_{inf}). It has also been observed that when a constant flux assumption is made in the DRAGON and WIMS8 code runs, the time evolutions of the k_{inf} from the two codes are similar. Additionally, since the initial k_{inf} is typically similar for the two codes, these trends suggest the need to evaluate the depletion chain data (fission product yields, effective energy released per fission [Q-values], decay constants, etc.). The Q-values are important to obtaining accurate flux levels in lattice and whole-core neutronics calculations; the flux level has a direct effect on the evolution of the fuel mass composition over depletion time, which in turn could affect the core reactivity.

In the ANL utilization of the DRAGON code, the depletion chain data are taken directly from the WIMS-ANL cross-section library used with the code. It has however been observed that different code libraries might have different Q-values. Table 8 is a comparison of the Q-values contained in the WIMS-ANL library (69- or 172-group) to those in the IAEA WIMS-D and WIMS8 libraries. It is observed that the latter libraries have identical Q-values, while the WIMS-ANL library values are different.

In order to evaluate the impact of the differences in the Q-values of Table 8, we have developed a procedure to modify manually the depletion chain data during a DRAGON run. This allows overriding the data contained in the cross section (binary-file) library. This procedure uses the “*descdepl*” option of the “*LIB:*” module of DRAGON. To first test this user defined depletion chain algorithm, the same data contained in the cross-section library has been collected and used in the DRAGON input deck. The results (eigenvalue and nuclide number densities) obtained with the code were then compared with those obtained using the depletion chain data in the cross-section library. If the algorithm is working well and the depletion data (i.e., fission product yields, decay constants, Q-values, etc) of the input deck are identical to the data in the library, the same results should be obtained. For this study, the WIMS-ANL library was used as the reference library and a homogeneous cell problem (“*TCWD08-burnup of homogeneous cell*”) of the DRAGON code package (manual) [2] was utilized.

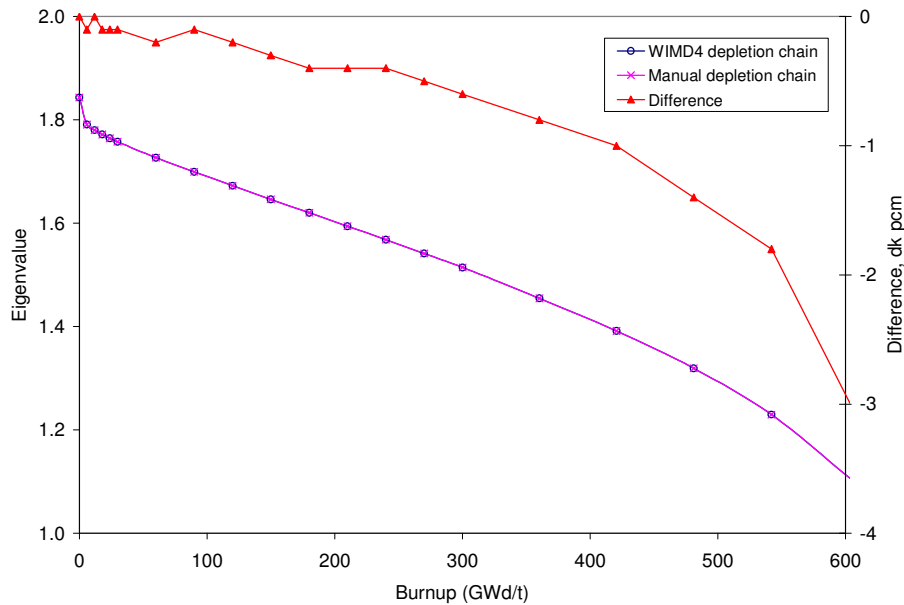


Figure 4. Comparison of Eigenvalues from Different Implementations of Depletion Chains.

Figure 4 shows the eigenvalues and differences between the two depletion cases. The result labeled “*WIMSD4 depletion chain*” denotes the case using DRAGON with the depletion chains specified in the library, while “*Manual depletion chain*” denotes the case using DRAGON with the depletion chains specified in the input deck. The eigenvalues (k_{inf}) obtained for the cases

are quite similar for the whole depletion period (maximum difference less than 3 pcm Δk). In addition, nearly identical number densities were observed. The results indicate that the user specified depletion chains for the DRAGON code are correctly specified and employed as input to the code.

To evaluate the impact of the different isotopic Q-values on the VHTR fuel element results, a depletion problem for the VHTR has been employed. The VHTR fuel element geometry is identical to that used in Section 2.2; the fuel kernel size is 350 μm and the fuel packing fraction in the graphite matrix is 0.289. The uranium enrichment of the fuel is 10% U-235. The VHTR fuel element eigenvalue letdown curves obtained with the DRAGON code using different libraries (and Q-values) are compared in Figure 5. It is observed that the results obtained with the IAEA WIMS-D libraries (using the same Q-values) are quite similar. However, as expected, that obtained using the WIMS-ANL library is different (lower) than the others, with the difference increasing with burnup. The smaller Q-values of the WIMS-ANL library compared to the others gives a higher flux level; thus the depletion rates of the fissile nuclides increase (or the k_{inf} decreases) with the depletion calculation.

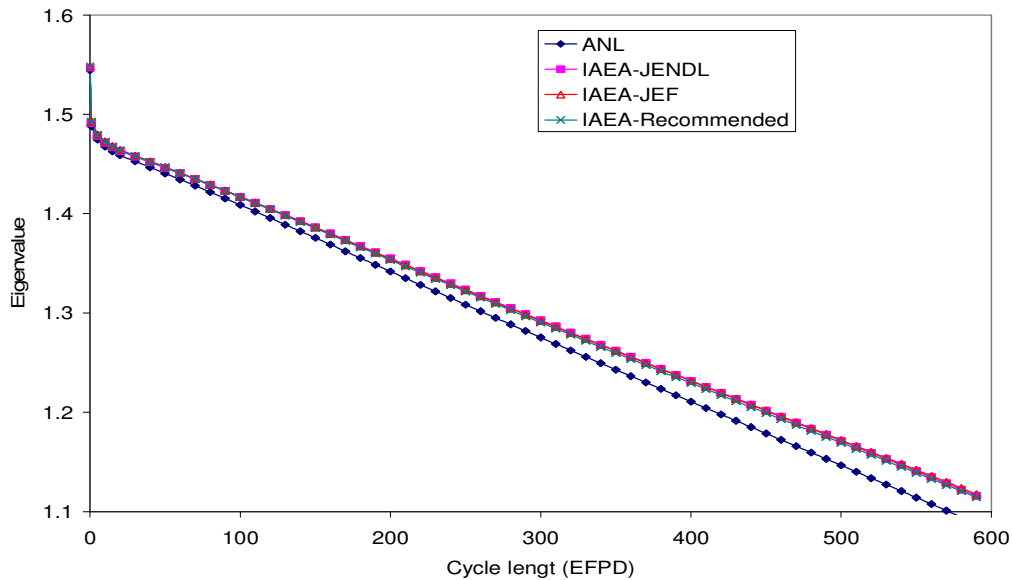


Figure 5. Depletion Results with Original Q-Values of Libraries.

When the Q-values of the WIMS-ANL library are replaced manually with those from the WIMS8 library, closer agreements in the depletion results are observed. This is shown in

Figure 6. Given the level of code and data qualification that has been done for the WIMS8 library in the U.K. program, it has been decided to use the WIMS8 library Q-values in future DRAGON depletion calculations.

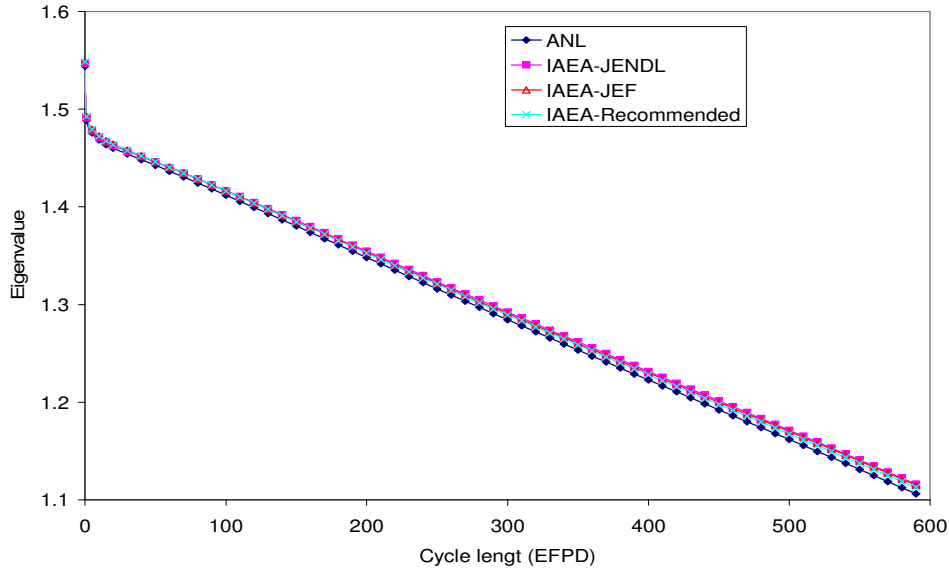


Figure 6. Depletion Results with Modified WIMS-ANL Q-Values.

3.3 Performance of WIMS-D Libraries

Another potential source of differences in the results obtained with different libraries is the cross section data contained in the libraries. Three numerical benchmark problems have been used to evaluate the differences in results obtained using the WIMS-ANL libraries and the IAEA WLUP libraries. The first two problems are infinite cell benchmarks, provided by the IAEA WLUP for testing the library implementation on different computers. These problems are designated ZPR7-a2 and BNW-Pu-a1. [6] These cases are both light water moderated lattice problems, but with different fuel compositions. The ZPR7-a2 problem is for a low-enriched uranium oxide fuel (uranium enrichment of 3.04%), while BNW-Pu-a1 is weapons-grade MOX fuel. Both cases are hexagonal lattice cell problems.

Calculation results for the infinite cell problems are summarized in Table 9. The MCNP4C calculations listed on the table have been used as reference solutions. Generally, the results obtained using the DRAGON code with the WIMS-ANL library are closer to the reference values (less than 400 pcm $\Delta\rho$), compared to those obtained with the IAEA WLUP

library (about 600 – 110 pcm). The MCNP4C and DRAGON/WIMS-ANL calculations used ENDF/B-VI based data, while as noted earlier, most of the IAEA WLUP library data are derived from JEF2.2 data files. Given the simple nature of the problems and the relatively large differences between the reference and DRAGON/WLUP-WIMS solutions, the differences have been attributed to either (1) differences between the ENDF/B-VI and JEF2.2 data or (2) differences in the cross section generation path (e.g., different assumptions in the NJOY code input used in the library creation process).

Table 9. Comparison of Eigenvalues for Infinite Cell Benchmarks.

Problem	Fuel	Library	Code	Eigenvalue*
ZPR7-a2	UO ₂	ENDF/B-VI	MCNP4C	1.26732 ± 0.0006
		ANL	DRAGON	1.27108 (233)
		IAEA	DRAGON	1.28499 (1085)
BNW-Pu-a1	MOX	ENDF/B-VI	MCNP	1.35787 ±0.0006
		ANL	DRAGON	1.35064 (-394)
		IAEA	DRAGON	1.34677 (-607)

*Value in parenthesis is difference between DRAGON and MCNP4C results in pcm Δp.

The third problem is for a VHTR fuel element; the same fuel element used for the study reported in Section 2.2. Again, reference solutions have been generated using the MCNP4C code. Additionally, results obtained with the WIMS8 code have been included for comparison.

Results of the calculations are summarized in Table 10. Eigenvalues obtained using the 69- and 172-group libraries are included. The following trends in results are observed. The DRAGON calculations show that the results are dependent on the number of energy groups. The results obtained with the 69-group cross section libraries are generally worse than those with the 172-group libraries, when compared to the reference MCNP4C solution. The use of the 172-group libraries results in less than 700 pcm over-prediction of the eigenvalue, while the 69-group libraries give a much higher difference (an additional ~300 pcm higher than the 172-group library results).

It is also observed that the WIMS8 calculation using JEF2.2 data gave the smallest difference compared to the MCNP4C reference solution; it under-predicts the eigenvalue by 122 pcm. This low difference may have arisen from the cancellation of errors between the code

solution effect and the library effect. Note again that the DRAGON calculations with the WLUP library (IAEA recommended library and the IAEA-JEF2.2 library) are predominantly based on JEF2.2 data. The use of these libraries result in the over-prediction of the eigenvalue and have higher differences relative to the MCNP4C results (an additionally 100 – 200 pcm) than those obtained with the WIMS-ANL library.

The magnitudes of some of these differences are not satisfactory and additional evaluations of the code and nuclear-data library are required.

Table 10. VHTR Assembly Eigenvalues with Different Libraries.

Code	Library	Group	Eigenvalue
MCNP4C	ENDF/B-VI	Continuous	1.53280 ± 0.00082
WIMS8	WIMS8 library	172	1.52993 (-122)
DRAGON	ANL WIMS-D library	172	1.54393 (470)
	IAEA recommended library		1.54850 (661)
	IAEA-JEF2.2 library		1.54814 (646)
	IAEA-JENDL3 library		1.54689 (594)
	ANL WIMS-D library	69	1.55198 (806)
	IAEA recommended library		1.55639 (989)
	IAEA-JEF2.2 library		1.55627 (984)
	IAEA-JENDL3 library		1.55461 (915)

*Value in parenthesis is difference between DRAGON and MCNP4C results in pcm $\Delta\rho$.

4. DRAGON CODE WORKSHOP

A workshop on the DRAGON code was organized in FY 2005 in order to familiarize some participants of the Gen IV and AFCI programs that have not used DRAGON with the code and to provide a refresher course for some of the other code users. This workshop was held at ANL, February 28 and March 1, 2005, and was attended by ~20 people from ANL, INL, ORNL, BNL, and KAERI (Korea). The format of the workshop was a technical exchange on the code. One of the code authors, Dr. Guy Marleau (Ecole Polytechnique de Montreal), gave a series of presentations. There were also opportunities for fruitful discussions between him and the participants, particularly concerning the application of the code for VHTR analysis.

The topics covered during the workshop included general code theory and features, cross section libraries to be used with the code, user instructions, and sample decks/problems. There were also discussions on the double heterogeneity and resonance treatments for coated fuel particles, transmutation and decay chains, modularity of code for staged calculations, and code performance compared to those of higher fidelity models/codes and experiments. Additional items included discussions on detailed code documentation and descriptions of advanced models and capabilities not in public domain (e.g., method of characteristics solution, homogenization and de-homogenization, parallel code version). A preliminary assessment of code deficiencies and necessary modifications that would make the code attractive for VHTR applications was also pursued.

There were positive feedbacks from the workshop participants. The code showed promise as one to be considered for use in the Gen IV program for the analysis of thermal systems. The code theories are also applicable to fast spectrum systems, but significant modifications would have to be made to the models used for resonance and slowing-down treatments. It is being planned to invite the DRAGON code developers to give another seminar in FY 2006 focusing more on the resonance and homogenizations treatments for the particulate fuels used in the advanced high temperature reactor systems.

The Workshop Agenda, list of participants, and the presentation material from the workshop are provided in Appendix A.

5. CONCLUSIONS AND RECOMMENDATIONS

A preliminary evaluation of the open-source lattice code DRAGON has been done to determine its performance in the analysis of the Very High Temperature Reactor (VHTR) fuel element designs. The activities performed under this project in FY 2005 were divided into three parts:

- DRAGON code was used to model the VHTR fuel elements of the helium-cooled and liquid-salt-cooled VHTR designs. Results were compared to those from another deterministic lattice code (WIMS8) and Monte Carlo code (MCNP). These results showed the promise of the DRAGON code. Some relatively large differences when the code results are compared to MCNP4C and WIMS8 results suggest that additional developmental work and validation might be necessary for bring the code to a level that it can be used routinely for VHTR analysis with great confidence.
- The nuclear data library currently used with the code was evaluated by comparing the code performance against those obtained when data libraries obtained from the IAEA WIMS-D4 Library Update Project (WLUP) are used. This step was also an evaluation of the WLUP libraries to investigate their potential for use in future DRAGON calculations. Differences were observed in the results obtained using the different libraries that could mostly be attributed to the data contained in the cross section libraries. Additional evaluations of these differences and their sources are recommended.
- A DRAGON workshop was held to discuss the code capabilities for modeling the VHTR. A code author visited ANL to deliver a two-day seminar on the DRAGON code to a group of reactor physics specialists from U.S. national laboratories. Future collaboration with the DRAGON code team is recommended to support the development of the code in order to provide features that make the code both accurate and more user-friendly for VHTR applications.

ACKNOWLEDGMENTS

This ANL work was supported by the U.S. Department of Energy, Office of Nuclear Energy, Science and Technology (VHTR element of the Generation IV Program) under Contract number W-31-109-Eng-38.

The contributions of Jessica Matheron, a 2005 summer intern in the Nuclear Engineering Division at ANL, are appreciated. She participated in the evaluation of the IAEA WIMS-D Library Update Project (WLUP) libraries.

REFERENCES

1. T. A. Taiwo, W. S. Yang, M. A. Smith, T. K. Kim, and H. S. Khalil, "Assessment of Reactor Physics Codes for Analysis and Design of the Very High Temperature Reactor (VHTR)," Argonne National Laboratory, August 31, 2005.
2. G. Marleau, et al, "A User Guide for DRAGON," Technical report IGE-174 Rev. 4, Ecole Polytechnique de Montréal, September 1998.
3. "WIMS – A Modular Scheme for Neutronics Calculations," User's Guide for Version 8, ANSWER/WIMS(99)9, The ANSWERS Software Package, AEA Technology.
4. T. K. Kim, W. S. Yang, M. A. Smith, T. A. Taiwo, and H. S. Khalil, "*Assessment of Monte Carlo and Deterministic Codes for Next Generation Nuclear Plant (NGNP) Core Modeling*," Gen-IV Report, Argonne National Laboratory (April 15, 2004).
5. T. K. Kim, T. A. Taiwo, and W. S. Yang, "Preliminary Neutronic Studies for the Liquid-Salt-Cooled Very High Temperature Reactor (LS-VHTR)," ANL-GenIV-052, Argonne National Laboratory, August 31, 2005.
6. WLUP internet reference: <http://www-nds.iaea.org/wimsd/downloads.htm>

APPENDIX A

**AGENDA, LIST OF PARTICIPANTS, AND PRESENTATION MATERIAL FOR
THE DRAGON WORKSHOP**

Argonne National Laboratory

February 28 and March 1, 2005

DRAGON Workshop

*Presentations by
Dr. Guy Marleau
Ecole Polytechnique de Montreal, Canada*

Monday February 28 – Tuesday March 1, 2005
Argonne National Laboratory
Argonne, Illinois 60439, U.S.A.

Agenda for DRAGON Workshop

Presentations by Dr. Guy Marleau, Ecole Polytechnique de Montreal

Monday February 28 – Tuesday March 1, 2005

Argonne National Laboratory

Argonne, Illinois 60439, U.S.A.

Monday:

- | | | |
|-------------|-----|----------------------------------|
| 9:00-12:00 | 1. | Presentation of DRAGON 3.04. |
| | 1.1 | General features. |
| | 1.2 | Cross section libraries. |
| | 1.3 | Geometry and tracking. |
| | 1.4 | Resonance self-shielding. |
| | 1.5 | Homogenization and condensation. |
| | 1.6 | Depletion calculations. |
| 14:00-17:00 | 2. | Status of DRAGON. |
| | 2.1 | Verification. |
| | 2.2 | Validation. |
| | 2.3 | Documentation. |
| | 2.4 | Using DRAGON. |

Tuesday:

- | | | |
|-------------|-----|--|
| 9:00-12:00 | 3 | DRAGON today and tomorrow. |
| | 3.1 | Advanced features in DRAGON. |
| | 3.2 | Current improvements in version 3.04 revision R. |
| | 3.3 | Limitations in DRAGON. |
| | 3.4 | What to expect in version 3.05. |
| | 3.5 | Towards DRAGON 4.00. |
| 13:00-14:00 | 4 | Discussion. |
| | 4.1 | Features required for Generation IV reactors analysis. |
| | 4.2 | Additional user's feedback. |

End of Workshop

- | | | |
|-------------|---|--|
| 14:00-16:00 | 5 | Meeting with ANL-NE Division Management. |
|-------------|---|--|

List of Participants for DRAGON Workshop

Presentations by Dr. Guy Marleau, Ecole Polytechnique de Montreal

Monday February 28 – Tuesday March 1, 2005

Argonne National Laboratory

Argonne, Illinois 60439, U.S.A.

Name	Phone Number	E-mail
ANL		
G. Aliberti	630 252 7258	aliberti@anl.gov
E. Hoffman	630 252 6683	ehoffman@anl.gov
S. J. Kim	630 252 1644	sjkim@ne.anl.gov
T. Kim	630 252 3213	tkkim@anl.gov
D. Naberezhnev	630 252 7402	dimitri@anl.gov
G. Palmiotti (Pino)	630 252 2858	GPalmiotti@anl.gov
M. Smith	630 252 3469	masmith@anl.gov
T. Taiwo	630 252 1387	Taiwo@anl.gov
W. Yang	630 252 9747	wyang@anl.gov
BNL		
A. Aronson	631 344 2606	aronson@bnl.gov
H. Ludewig	631 344 2624	ludewig@bnl.gov
INL		
H. Gougar	208 526 2760	hans.gougar@inl.gov
D. Nigg	208 526 7627	david.nigg@inl.gov
A. Ougouag	208 526 7659	abderrafi.ougouag@inl.gov
W. Yoon	208 526 1879	woo.yoon@inl.gov
ORNL		
M. DeHart	865 576 3468	DeHartMD@ornl.gov
K. Clarno	865 241 1894	clarnokt@ornl.gov
KAERI		
K. S. Kim		kimks@kaeri.re.kr



Description of DRAGON 3.04

G. Marleau

Institut de génie nucléaire
École Polytechnique de Montréal



Description of DRAGON 3.04

1

Contents:

1. General features.
2. Cross-section libraries.
3. Geometry and tracking.
4. Resonance self-shielding.
5. Homogenization and condensation.
6. Depletion calculations.



General features

1

DRAGON is a lattice code:

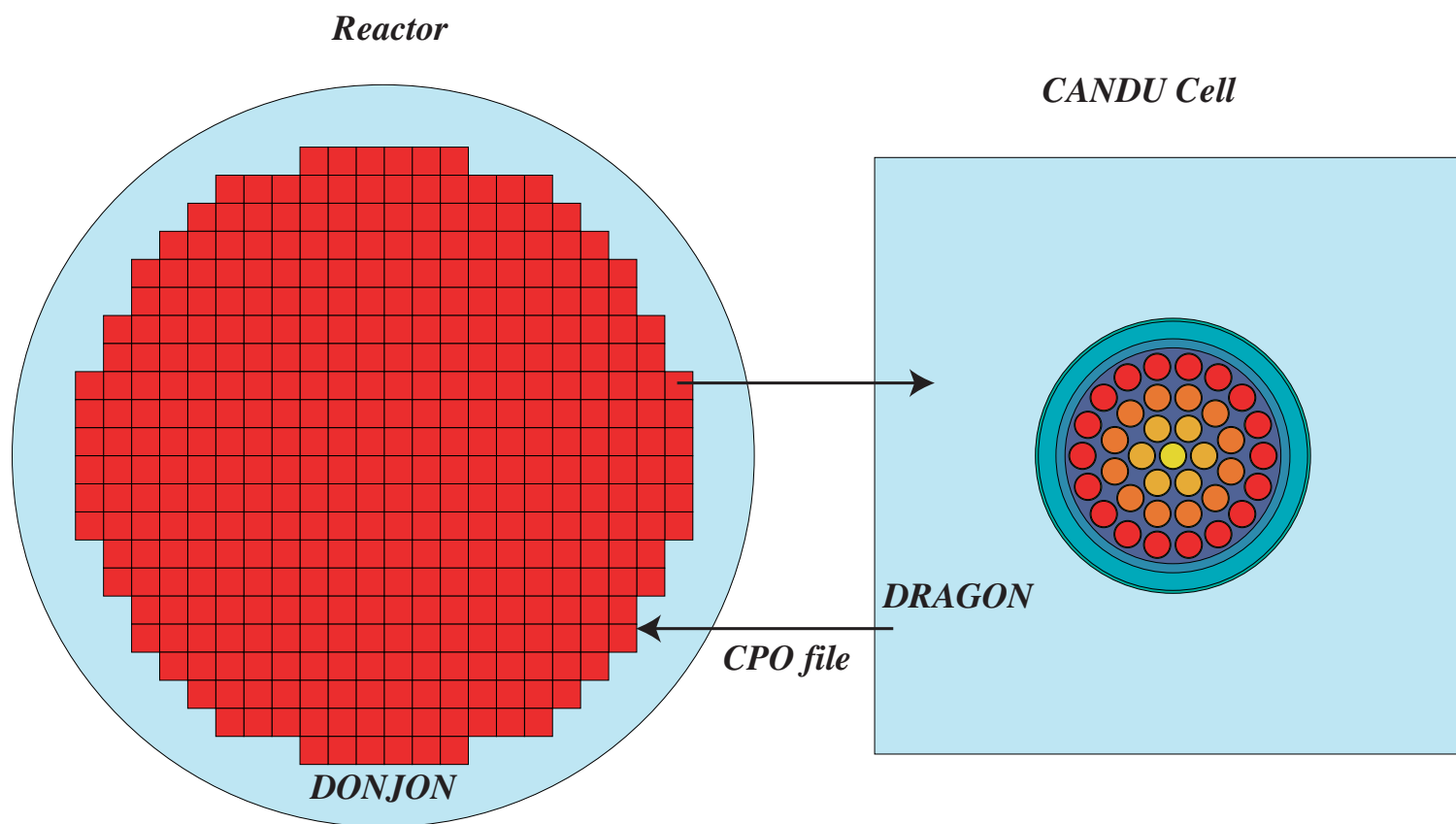
- It should be used to compute the neutron flux distribution inside assemblies of heterogeneous cells.
- It should use the computed neutron flux distribution to produce the homogenized and condensed nuclear properties that will be used in finite reactor calculations.
- It should solve the isotopic depletion equations and predict the concentration of each isotope in the cell as a function of time.



General features

2

CANDU example (reactor and cell):

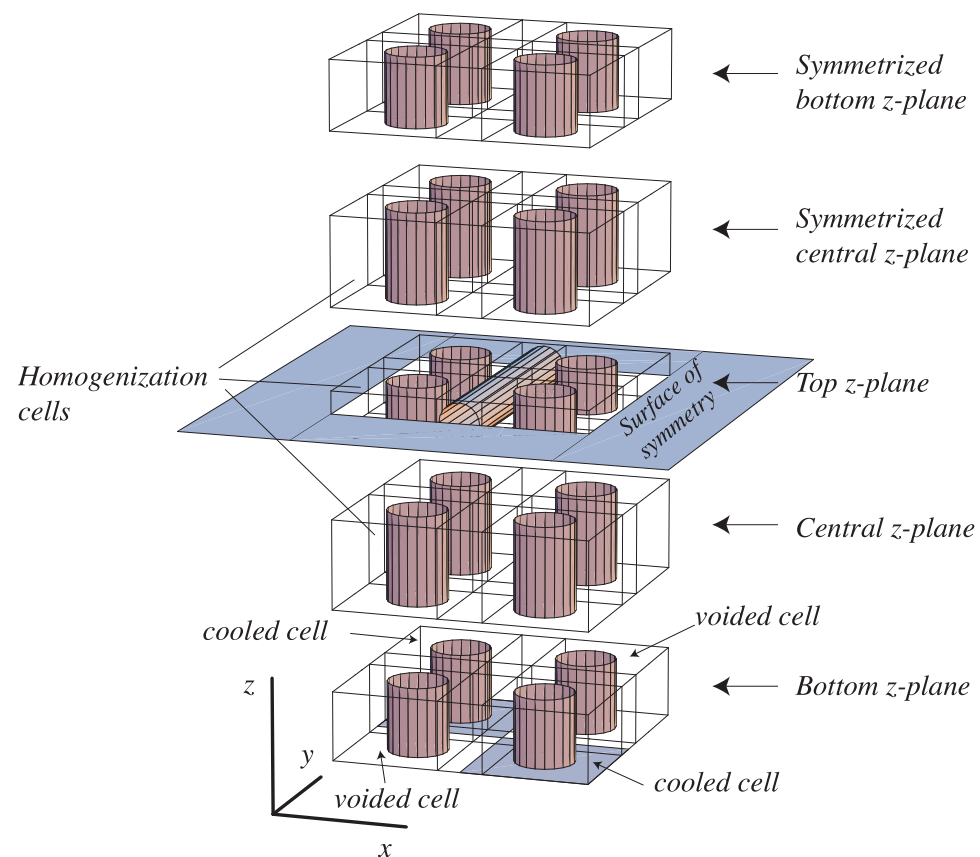




General features

3

CANDU example (reactivity devices):

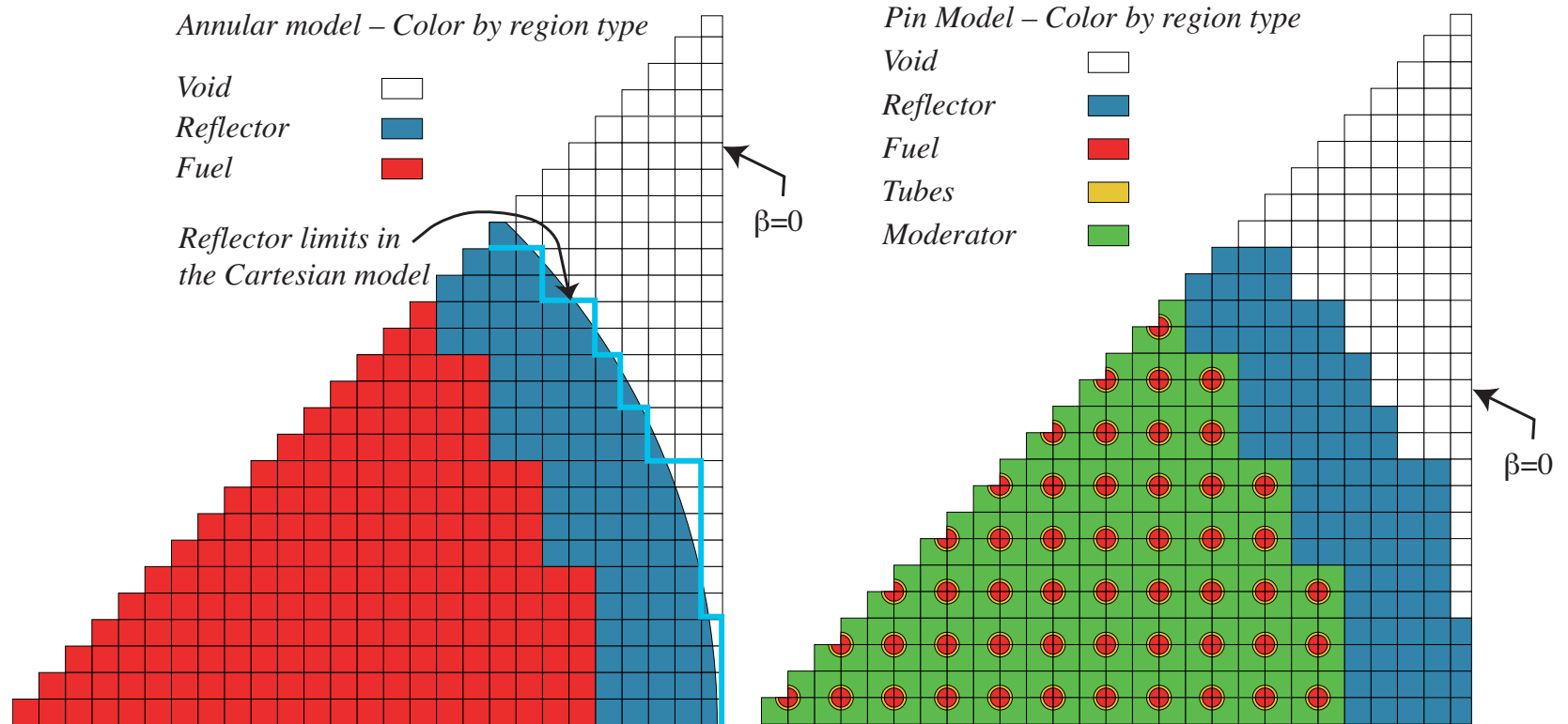




General features

4

CANDU core in 2-D:





General features

5

CANDU reactor:

- 79 different physical regions in each fuel cell.
- 380 fuel cells in the 2-D reactor plane.
- 12 planes of fuel in the 3-D reactor.
- 3-D fuel bundles are not uniform along the axis (end-regions).
- Reactivity and shutdown devices placed at various location in the core.
- Over 700,000 different physical regions in the core.

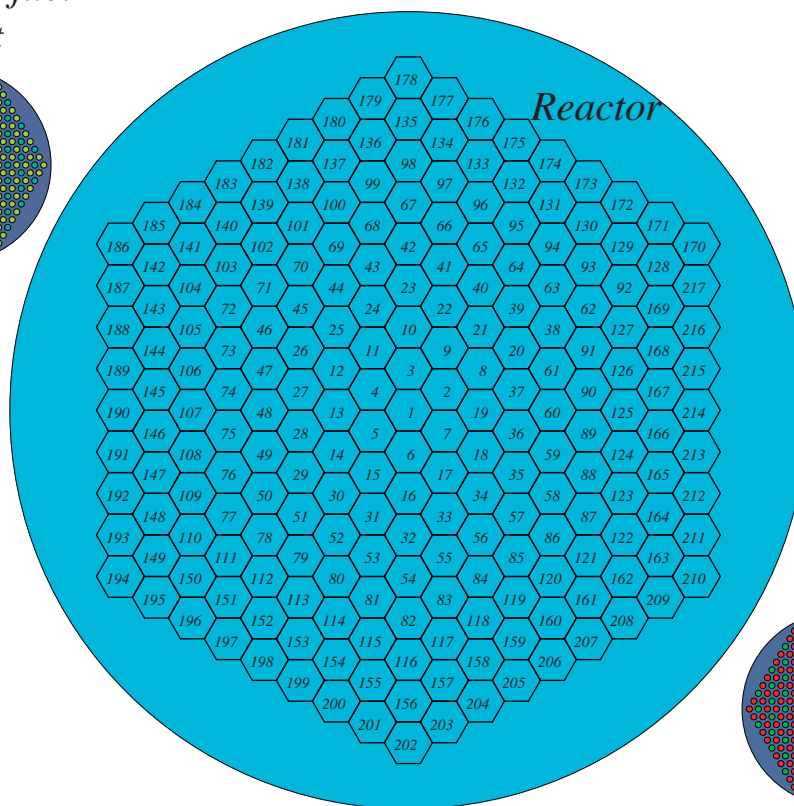
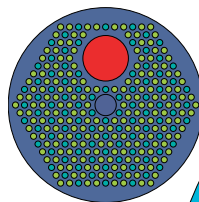


General features

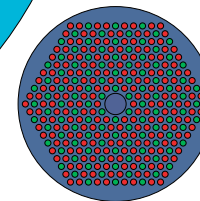
6

Prismatic modular reactor example (reactor and assemblies):

Control fuel element



Standard fuel assembly





General features

7

2–D assembly calculations using DRAGON:

- Assume assembly is uniform and infinite in the axial direction.
- Use reflective or periodic boundary conditions at assembly limits.
- Extract multigroup cell properties from microscopic cross section library.
- Self-shield the cross section for resonant isotopes.



General features

8

2–D assembly calculations using DRAGON (continue):

- Perform fine mesh multigroup transport calculation inside 2–D assembly.
- Perform isotopic depletion calculations.
- Generate burnup dependent fewgroup (multigroup) homogenized (or partly homogenized) assembly properties.



General features

9

3–D supercell (assembly) calculations using DRAGON:

- Assume reactor is made up of an infinite number of identical supercell.
- Use reflective or periodic boundary conditions at supercell limits.
- Extract multigroup, partly homogenized assembly properties from 2–D assembly calculation (EDI : database).
- Perform fine mesh multigroup transport calculation inside 3–D supercell.
- Generate fewgroup homogeneous properties for 3-D supercell.



General features

10

2–D or 3–D reactor calculations using DRAGON:

- Use burnup dependent fewgroups properties from assembly calculation (EDI : database).
- Use vacuum boundary conditions at reactor limits.
- Perform fine mesh multigroup transport calculation inside 2–D or 3–D reactor model.
- Generate fission power distribution in the reactor.



General features

11

Using Version 3.04 of DRAGON, one can:

- Solve the neutron transport equation using the CP method in 1, 2 and 3-D geometry.
- Solve the neutron transport equation using the method of cyclic characteristics in 2-D geometry.
- Solve the isotopic depletion equations for isotopic concentration.



General features

12

One can also:

- Process multigroup microscopic cross section libraries including self-shielding.
- Process multigroup macroscopic cross section libraries generated by DRAGON or other codes (WIMS-AECL and HELIOS have been coupled to DRAGON for CANDU applications).
- Generate the fewgroup/homogenized information required for finite reactor calculations (DRAGON EDI: format and DONJON CPO: and FBM: formats are supported).



General features

13

The multigroup static neutron transport equation:

$$\left[\vec{\Omega} \cdot \vec{\nabla} + \Sigma^g(\vec{r}) + D^g(\vec{r})B^2 \right] \Phi^g(\vec{r}, \vec{\Omega}) = Q_s^g(\vec{r}, \vec{\Omega}) + \frac{1}{k} Q_f^g(\vec{r}) + Q_e^g(\vec{r})$$

Scattering source $Q_s^g(\vec{r}, \vec{\Omega})$:

$$Q_s^g(\vec{r}, \vec{\Omega}) = \sum_{h=1}^G \int d^2\Omega' \Sigma_s^{hg}(\vec{r}, \vec{\Omega}' \rightarrow \vec{\Omega}) \Phi^h(\vec{r}, \vec{\Omega}')$$

Fission source $Q_f^g(\vec{r})$:

$$Q_f^g(\vec{r}) = \chi^g \sum_{h=1}^G \nu \Sigma_f^h(\vec{r}) \int d^2\Omega' \Phi^h(\vec{r}, \vec{\Omega}')$$



General features

14

- The multigroup macroscopic cross sections are defined as:

$$\Sigma_{m,x}^g(\vec{r}, \vec{\Omega}) = \sum_I N_{m,I}(\vec{r}) \sigma_{I,x}^g(\vec{\Omega})$$

for mixture m that contains the isotopes I .

- The multigroup microscopic cross sections are extracted from a cross-section library (many formats supported).
- The multigroup microscopic cross sections can be self-shielded.



General features

15

Transport equation in the absence of external sources ($Q_e^g(\vec{r}) = 0$) is an eigenvalue problem:

1. k eigenvalue with imposed leakage ($D^g(\vec{r})B^2$ fixed):
 - k indicates how the fission rate should be modified to make the system critical (reach a non-trivial solution to the transport equation).
2. Buckling eigenvalue with imposed k :
 - $D^g(\vec{r})B^2$ represent the amount of leakage required to make the system critical.



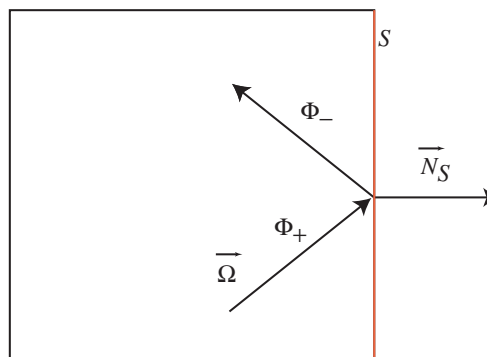
General features

16

Boundary conditions are required to close the system:

● Albedo conditions:

$$\Phi_{-}(\vec{r}_S, \vec{\Omega} - 2(\vec{N}_S \cdot \vec{\Omega})) = \beta(\vec{r}_S, \vec{\Omega}) \Phi_{+}(\vec{r}_S, \vec{\Omega})$$



- $\beta(\vec{r}_S, \vec{\Omega}) = 0$ for vacuum (void) BC.
- $\beta(\vec{r}_S, \vec{\Omega}) = 1$ for reflective BC.



General features

17

The static multigroup neutron transport equation (integral form) for the angular flux:

$$\begin{aligned}\phi^g(\vec{r}, \vec{\Omega}) = & e^{-\tau^g(R_S)} \Phi_-^g(\vec{r}'_S, \vec{\Omega}) \Theta(\vec{r}, \vec{r}'_S, \vec{\Omega}) \\ & + \int_0^R e^{-\tau^g(R')} Q^g(\vec{r}', \vec{\Omega}) \Theta(\vec{r}, \vec{r}', \vec{\Omega}) dR'\end{aligned}$$

with $R = |\vec{r} - \vec{r}'|$ and:

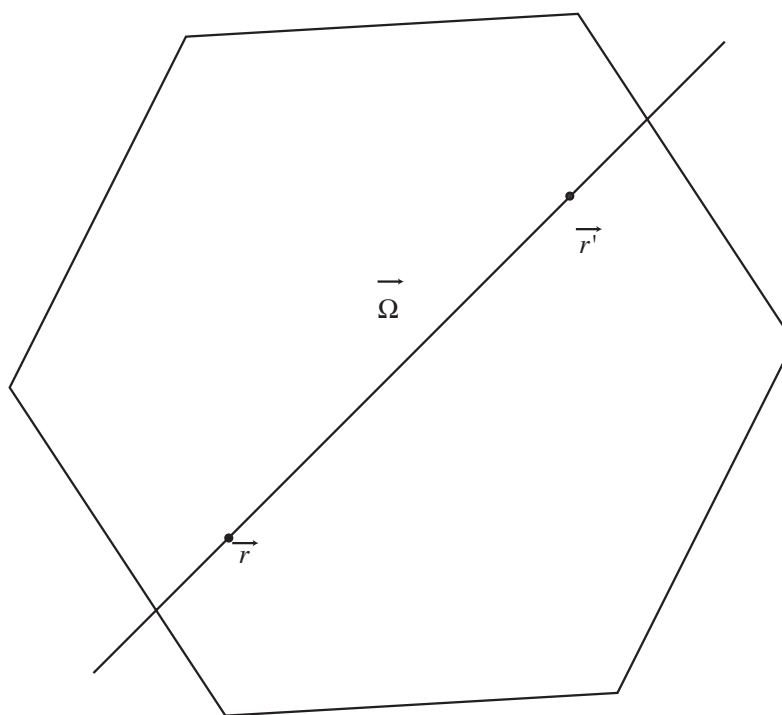
$$\Theta(\vec{r}, \vec{r}', \vec{\Omega}) = \begin{cases} 1 & \text{for } \vec{r}' = \vec{r} - R\vec{\Omega} \\ 0 & \text{otherwise} \end{cases}$$



General features

18

Example of a neutron path for the integral transport equation:





General features

19

- Optical path $\tau^g(R)$:

$$\tau^g(R) = \int_0^R \Sigma^g(\vec{r} - R'\vec{\Omega}) dR'$$

- Neutron source $Q^g(\vec{r}, \vec{\Omega})$ includes scattering, fission and fixed sources:

$$Q^g(\vec{r}, \vec{\Omega}) = Q_s^g(\vec{r}, \vec{\Omega}) + \frac{1}{k} Q_f^g(\vec{r}) + Q_e^g(\vec{r})$$

- $\Phi_-^g(\vec{r}'_S, \vec{\Omega})$ is incoming angular flux at surface \vec{r}'_S .



General features

20

Boundary conditions are required to close the system:

- For cell or assembly isolated in space one uses vacuum BC:

$$\Phi_{-}^g(\vec{r}_S, \vec{\Omega}) = 0$$



General features

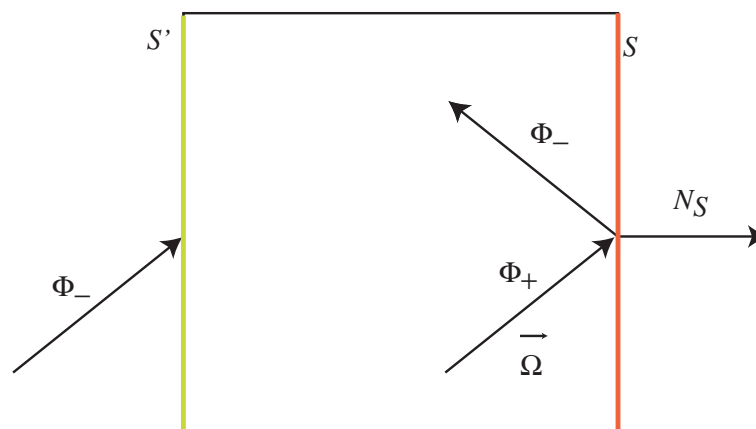
21

- For cell or assembly repeating to infinity albedo conditions or periodic BC are considered:

$$\Phi_{-}(\vec{r}_S, \vec{\Omega} - 2(\vec{N}_S \cdot \vec{\Omega})) = \beta(\vec{r}_S, \vec{\Omega})\Phi_{+}(\vec{r}_S, \vec{\Omega})$$

$$\Phi_{-}(\vec{r}'_S, \vec{\Omega}) = \Phi_{+}(\vec{r}_S, \vec{\Omega})$$

with $\Phi_{+}^g(\vec{r}'_S, \vec{\Omega})$ the outgoing angular flux at surface \vec{r}_S .





General features

22

Approaches used in DRAGON to solve the multigroup neutron transport equation:

- Collision probability method.
- Method of characteristics.



General features

23

Approximations for CP and MOC:

- Divide domain into N_V regions of volume V_i with constant and isotropic sources and cross sections:

$$\Sigma^g(\vec{r}) = \Sigma_j^g \quad \text{for } \vec{r} \in V_j$$

$$Q^g(\vec{r}, \vec{\Omega}) = \frac{1}{4\pi} q^g(\vec{r}) = q_j^g \quad \text{for } \vec{r} \in V_j$$

- Divide the external boundary S into N_S surfaces of area S_α with constant angular flux:

$$\Phi_-^g(\vec{r}_S, \vec{\Omega}) = \frac{1}{4\pi} \phi_{\alpha,-}^g$$



General features

24

For CP, integrate transport equation over regions and surfaces:

$$\phi_i^g = \frac{1}{V_i} \int_{V_i} d^3r \int d^2\Omega \phi^g(\vec{r}, \vec{\Omega})$$

$$\phi_{+,\alpha}^g = \frac{1}{S_\alpha} \int_{S_\alpha} d^2r \int d^2\Omega (\vec{\Omega} \cdot \vec{N}_-) \phi^g(\vec{r}_S, \vec{\Omega})$$

The CP transport equations are:

$$\phi_i^g = \sum_{\alpha=1}^{N_S} p_{i\alpha}^g \phi_{-,\alpha}^g + \sum_{j=1}^{N_V} p_{ij}^g q_j^g$$

$$\phi_{+,\alpha}^g = \sum_{\beta=1}^{N_S} p_{\alpha\beta}^g \phi_{-,\beta}^g + \sum_{j=1}^{N_V} p_{\alpha j}^g q_j^g$$



General features

25

Four types of probabilities:

$$\tilde{p}_{ij}^g = V_i p_{ij}^g = \int_{V_i} \int_{V_j} \frac{e^{-\tau^g(R)}}{4\pi R^2} \Theta_i \Theta_j d^3 r' d^3 r$$

$$\tilde{p}_{i\alpha}^g = V_i p_{i\alpha}^g = \int_{V_i} \int_{S_\alpha} \frac{e^{-\tau^g(R_S)}}{4\pi R_S^2} (\vec{\Omega} \cdot \vec{N}_-) \Theta_i \Theta_\alpha d^3 r' d^2 r$$

$$\tilde{p}_{\alpha i}^g = \frac{S_\alpha}{4} p_{\alpha i}^g = \int_{S_\alpha} \int_{V_i} \frac{e^{-\tau^g(R)}}{4\pi R^2} (\vec{\Omega} \cdot \vec{N}_+) \Theta_\alpha \Theta_i d^2 r' d^3 r$$

$$\begin{aligned} \tilde{p}_{\alpha\beta}^g = \frac{S_\alpha}{4} p_{\alpha\beta}^g &= \int_{S_\alpha} \int_{S_\beta} \frac{e^{-\tau^g(R_S)}}{4\pi R_S^2} (\vec{\Omega} \cdot \vec{N}_-) (\vec{\Omega} \cdot \vec{N}_+) \\ &\times \Theta_\alpha \Theta_\beta d^2 r d^2 r' \end{aligned}$$

These will be evaluated numerically.



General features

26

For the method of characteristics we will use again the integral transport equation and assume:

- Divide domain into N_V regions of volume V_i with constant and isotropic sources and cross sections:

$$\Sigma^g(\vec{r}) = \Sigma_j^g \quad \text{for } \vec{r} \in V_j$$

$$Q^g(\vec{r}, \vec{\Omega}) = \frac{1}{4\pi} q^g(\vec{r}) = q_j^g \quad \text{for } \vec{r} \in V_j$$

- Divide the external boundary S into N_S surfaces of area S_α with constant angular flux:

$$\Phi_-^g(\vec{r}_S, \vec{\Omega}) = \frac{1}{4\pi} \phi_{\alpha,-}^g$$



General features

27

- The sources $q_i(\vec{\rho}_k)$ are constant on each line segment $\vec{r} = \vec{r}' + t\vec{\Omega}$ crossing the domain in direction $\vec{\Omega}$.
- The incoming flux on a point \vec{r}_S on surface α is assumed known and given by $\psi_{\alpha,-}(\vec{\rho}_k, \vec{\Omega})$.



General features

28

The average flux inside each region will be given by:

$$\begin{aligned} V_i \phi_i = & \int \frac{d^2\Omega}{4\pi} \int d^2\rho \sum_{j=1}^{i-1} q_j(\vec{\rho}) \left(\frac{1 - e^{-\tau_j(\vec{\rho})}}{\Sigma_j} \right) \\ & \times \left[e^{-\sum_{l=j+1}^{i-1} \tau_l(\vec{\rho})} \right] \left(\frac{1 - e^{-\tau_i(\vec{\rho})}}{\Sigma_i} \right) \\ & + \int \frac{d^2\Omega}{4\pi} \int d^2\rho q_i(\vec{\rho}) \left(\frac{\tau_i(\vec{\rho}) - 1 + e^{-\tau_i(\vec{\rho})}}{\Sigma_i^2} \right) \end{aligned}$$

The $\frac{d^2\Omega}{4\pi}$ and $\int d^2\rho$ for MOC are evaluated numerically using the same procedure as for CP method.



General features

29

Isotopic depletion equations:

$$\frac{dN_I(t)}{dt} = \sum_J S_I^J(t) - N_I(t) \left(\int dE \sigma_{I,a}(E) \phi(E, t) + \lambda_I \right)$$

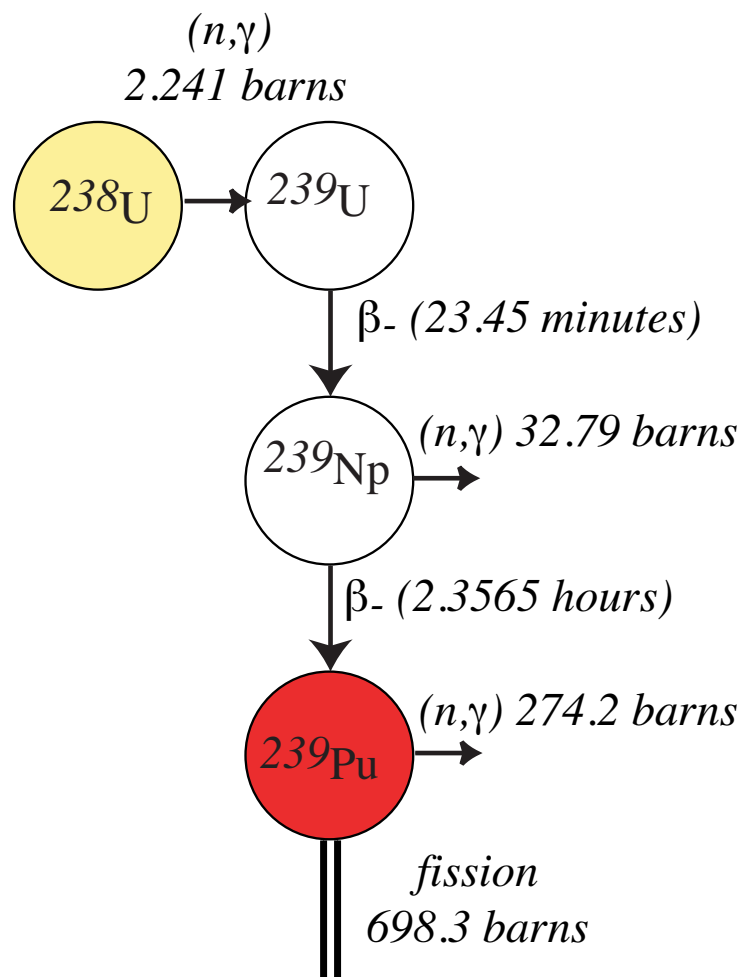
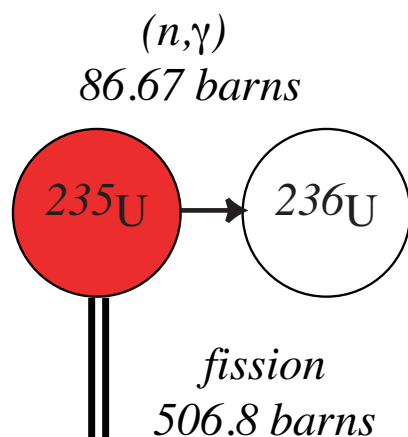
- λ_I is the decay rate of I .
- $\sigma_{I,a}(E)$ is the neutron absorption cross section for I .
- $S_I(t)$ is the production rate for isotope I from isotope J .
- Initial concentrations $N_I(t_0)$ required.



General features

30

Example of simplified decay chains for ^{235}U and ^{238}U :



● Fertile isotope
● Fissile isotope



General features

31

- Decay chain:
 - Available on some libraries.
 - Can be provided manually.
- Flux normalization options available:
 - Constant flux burnup.
 - Constant power burnup.
 - Out of core depletion.
- Includes an isotope saturation model.



Cross-section libraries

1

Two types of multigroup cross-section database can be accessed by DRAGON:

● MACROLIB:

- Contains the multigroup macroscopic cross-section associated with each mixture.
- Generally created by MAC : or EDI : modules.
- Required when macroscopic cross section are used (CP calculations module, flux solution module, editing module, etc.).



Cross-section libraries

2

● MICROLIB:

- Contains the multigroup microscopic cross-section associated with each isotope at a fixed temperature and dilution (self-shielded cross sections).
- Contains the depletion chain.
- Includes a MACROLIB.
- Generally created by `LIB:` or `EDI:` modules.
- Required when microscopic cross section are used (depletion calculations, self-shielding calculation, etc).



Cross-section libraries

3

Minimum cross-section requirements for each mixture m in a MACROLIB:

- The multigroup total cross section Σ_m^g .
- The isotropic component of the multigroup scattering cross section $\Sigma_{m,s,0}^{h \rightarrow g}$ defined as:

$$\Sigma_{m,s,0}^{h \rightarrow g} = \int_{4\pi} d^2\Omega^2 \Sigma_{m,s}^{h \rightarrow g}(\vec{\Omega}' \rightarrow \vec{\Omega}) P_0(\vec{\Omega}' \cdot \vec{\Omega})$$



Cross-section libraries

4

For multiplicative problems one also needs:

- The product of the average neutron emitted per fission with the multigroup fission cross section $\nu\Sigma_{m,f}^g$.
- The multigroup fission spectrum χ_m^g .

For a fixed source problem (multiplicative or not):

- The external neutron source q_m^g .



Cross-section libraries

5

Additional cross sections:

- The linearly isotropic component of the multigroup scattering cross section $\Sigma_{m,s,1}^{h \rightarrow g}$.

$$\Sigma_{m,s,1}^{h \rightarrow g} = \int_{4\pi} d^2\Omega^2 \Sigma_{m,s}^{h \rightarrow g}(\vec{\Omega}' \rightarrow \vec{\Omega}) P_1(\vec{\Omega}' \cdot \vec{\Omega})$$

- Required only if B_1 leakage method is used.



Cross-section libraries

6

- The transport correction $\Sigma_{m,\text{tc}}^g$.
- The transport calculations can be performed using transport corrected total ($\tilde{\Sigma}_m^g$) and scattering ($\tilde{\Sigma}_{m,s,0}^{h \rightarrow g}$) cross sections:

$$\tilde{\Sigma}_m^g = \Sigma_m^g - \Sigma_{m,\text{tc}}^g$$

$$\tilde{\Sigma}_{m,s,0}^{h \rightarrow g} = \Sigma_{m,s,0}^{h \rightarrow g} - \delta^{gh} \Sigma_{m,\text{tc}}^g$$

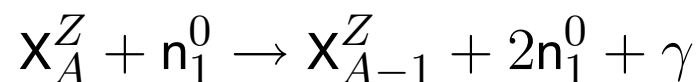
- Takes partially into account the linearly anisotropic scattering contributions.



Cross-section libraries

7

For contribution of multi-neutron production reactions such as:



- The scattering cross section (microscopic and macroscopic) are corrected to take into account this effect:

$$\tilde{\sigma}_{I,s,0}^{h \rightarrow g} = \sigma_{I,s,0}^{h \rightarrow g} + 2\delta^{gh} \sigma_{I,(n,2n)}^g$$

$$\tilde{\Sigma}_{m,s,0}^{h \rightarrow g} = \Sigma_{m,s,0}^{h \rightarrow g} + 2\delta^{gh} \Sigma_{m,(n,2n)}^g$$



Cross-section libraries

8

MACROLIB can be created:

- From the input file using the `MAC :` module.
- From a GOXS file using the `MAC :` module.
- From another MACROLIB using the `MAC :` module.
- From a basic microscopic cross section library using the `LIB :` module.
- From a MICROLIB using the `LIB :` module.
- From MICROLIB or a MACROLIB using the `EDI :` module.



Cross-section libraries

9

MACROLIB can be updated using:

- The **MAC**: module for direct cross section perturbation.
- The **LIB**: module for perturbation of the isotopic contents of a mixture.
- The **EVO**: module when the isotopic composition of the mixture has changed.



Cross-section libraries

10

Basic microscopic libraries that can be accessed in DRAGON:

- WIMS–AECL format.
- MATXS format.
- WIMD-D4 format.
- DRAGLIB format.



Cross-section libraries

11

WIMS-AECL microscopic cross section libraries:

- Direct access format only.
- Automatic selection of self-shielded isotopes.
- Automatic depletion chains.
- Winfrith 69 group library.
- ENDF/B-V based 89 group library.
- ENDF/B-VI based 89 group library.



Cross-section libraries

12

WIMSD4 microscopic cross section libraries:

- Sequential binary format.
- Manual selection of self-shielded isotopes.
- Automatic depletion chains.
- WIMSKAL 69 group library.
- ARGONNE ENDF/B-VI based 69 and 172 group libraries.
- IAEA WLUP Libraries 69 and 172 group libraries.
<http://www-nds.iaea.org/wimsd/downloads.htm>



Cross-section libraries

13

MATXS microscopic cross section libraries:

- Sequential binary format.
- Manual selection of self-shielded isotopes.
- No depletion chains.
- ENDF/B-V based 69 group library in NJOY-89 MATXS format.
- ENDF/B-V based 70 group library in NJOY-91 MATXS format.



Cross-section libraries

14

DRAGLIB microscopic cross section libraries:

- Direct access format (XSM file).
- Automatic selection of self-shielded isotopes.
- Automatic depletion chains.
- Winfrith 69 group library (a version of the library generated from the WIMS-AECL 69 groups library).



Cross-section libraries

15

Contents of the MICROLIB:

- An isotope on the MICROLIB refers to an isotope on the basic microscopic cross section library at a specified temperature and dilution.
- For each isotope, the LIB: module extracts the cross section from the basic library, interpolates in temperature and dilution and stores the information on the MICROLIB.
- The LIB: module initially assumes that the isotopes are at infinite dilution.
- The SHI: can be use to update the dilution cross section for resonance self-shielding based on the improved Stamm'ler method.



Cross-section libraries

16

Depletion chain

- The MICROLIB can also contain a depletion chain.
- This chain can be extracted from the basic cross section library (WIMS-AECL, WIMS-D4 and DRAGLIB).
- The chain can also be defined manually in the LIB: module.
- When the chain is defined manually, it must be consistent with the isotopic cross section information available on the MICROLIB.



Cross-section libraries

17

MICROLIB can be created:

- From a basic microscopic cross section library using the `LIB:` module.
- From another MICROLIB using the `LIB:` module.
- From another MICROLIB using the `EDI:` module.



Cross-section libraries

18

MICROLIB can be updated using:

- The **LIB**: module for manual perturbation of the isotopic contents of a mixture, dilution or temperature of an isotope.
- The **EVO**: module when the isotopic composition of the mixture has changed.
- The **SHI**: for self-shielding calculation.



Geometry and tracking

1

Approximations for CP methods:

- Divide domain into N_V regions of volume V_i with constant and isotropic sources and cross sections:

$$\Sigma^g(\vec{r}) = \Sigma_j^g \quad \text{for } \vec{r} \in V_j$$

$$Q^g(\vec{r}, \vec{\Omega}) = \frac{1}{4\pi} q^g(\vec{r}) = q_j^g \quad \text{for } \vec{r} \in V_j$$

- Divide the external boundary S into N_S surfaces of area S_α with constant angular flux:

$$\Phi_-^g(\vec{r}_S, \vec{\Omega}) = \frac{1}{4\pi} \phi_{\alpha,-}^g$$



Geometry and tracking

2

Integrate transport equation over regions and surfaces:

$$\phi_i^g = \frac{1}{V_i} \int_{V_i} d^3r \phi^g(\vec{r})$$

$$\phi_{+,\alpha}^g = \frac{1}{S_\alpha} \int_{S_\alpha} d^2r \phi^g(\vec{r}_S)$$

The CP transport equations are:

$$\phi_i^g = \sum_{\alpha=1}^{N_S} p_{i\alpha}^g \phi_{-,\alpha}^g + \sum_{j=1}^{N_V} p_{ij}^g q_j^g$$

$$\phi_{+,\alpha}^g = \sum_{\beta=1}^{N_S} p_{\alpha\beta}^g \phi_{-,\beta}^g + \sum_{j=1}^{N_V} p_{\alpha j}^g q_j^g$$



Geometry and tracking

3

Four types of probabilities:

$$\tilde{p}_{ij}^g = V_i p_{ij}^g = \int_{V_i} \int_{V_j} \frac{e^{-\tau^g(R)}}{4\pi R^2} \Theta_i \Theta_j d^3 r' d^3 r$$

$$\tilde{p}_{i\alpha}^g = V_i p_{i\alpha}^g = \int_{V_i} \int_{S_\alpha} \frac{e^{-\tau^g(R_S)}}{4\pi R_S^2} (\vec{\Omega} \cdot \vec{N}_-) \Theta_i \Theta_\alpha d^3 r' d^2 r$$

$$\tilde{p}_{\alpha i}^g = \frac{S_\alpha}{4} p_{\alpha i}^g = \int_{S_\alpha} \int_{V_i} \frac{e^{-\tau^g(R)}}{4\pi R^2} (\vec{\Omega} \cdot \vec{N}_+) \Theta_\alpha \Theta_i d^2 r' d^3 r$$

$$\begin{aligned} \tilde{p}_{\alpha\beta}^g = \frac{S_\alpha}{4} p_{\alpha\beta}^g &= \int_{S_\alpha} \int_{S_\beta} \frac{e^{-\tau^g(R_S)}}{4\pi R_S^2} (\vec{\Omega} \cdot \vec{N}_-) (\vec{\Omega} \cdot \vec{N}_+) \\ &\times \Theta_\alpha \Theta_\beta d^2 r d^2 r' \end{aligned}$$



Geometry and tracking

4

Symmetry relations:

$$V_i p_{ij}^g = V_j p_{ji}^g$$

$$4V_i p_{i\alpha}^g = S_\alpha p_{\alpha i}^g$$

$$S_\alpha p_{\alpha\beta}^g = S_\beta p_{\beta\alpha}^g$$

Conservation properties:

$$\sum_{\alpha=1}^{N_\alpha} p_{i\alpha}^g + \sum_{j=1}^{N_j} p_{ij}^g \Sigma_j^g = 1$$

$$\sum_{\beta=1}^{N_\beta} p_{\alpha\beta}^g + \sum_{i=1}^{N_i} p_{\alpha i}^g \Sigma_i = 1$$



Geometry and tracking

5

Surface flux approximation:

- Incoming angular flux on outer surfaces assumed to be independent of $\vec{\Omega}$.

Comments

- Outgoing angular flux on outer surfaces integrated over $\vec{\Omega}$.
- Angular flux not used at region interfaces.
- Approximation for incoming angular flux exact for vacuum BC.
- Approximation for incoming angular flux leads to large errors in surface flux when flux is not isotropic.

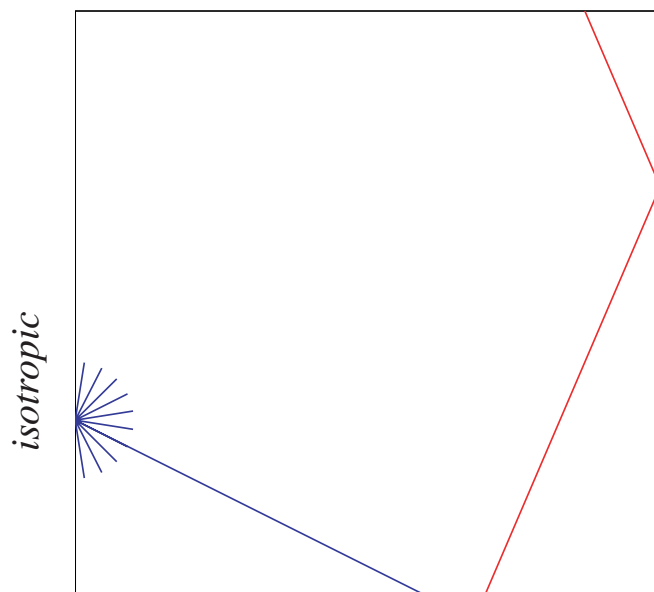


Geometry and tracking

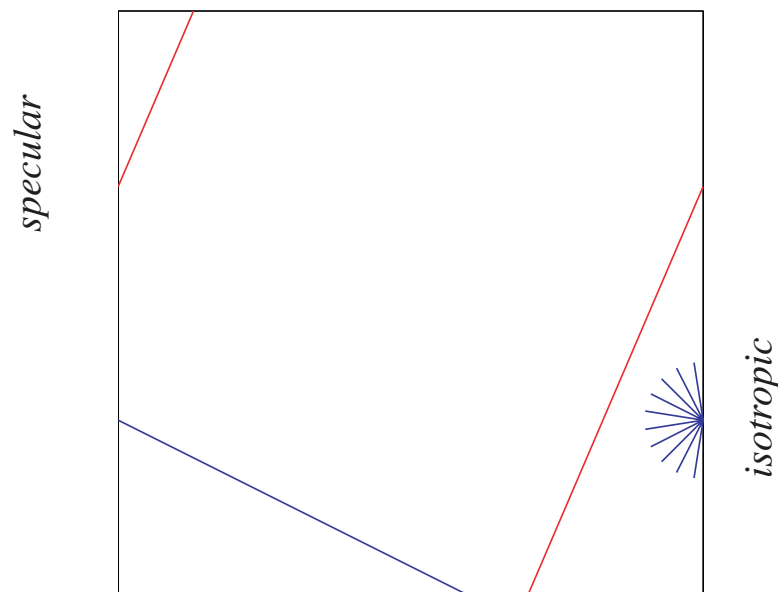
6

Illustration of approximate BC:

Reflection



Translation





Geometry and tracking

7

Recommendations to reduce errors due to approximate use of BC for a fixed direction:

- No special treatment for 2 vacuum BC.
- Unfold cell once for 1 vacuum and 1 reflection BC.
- Multiply unfold cell for 2 reflection or periodic cell, apply approximate BC on final surfaces and consider results in cell located far from these surfaces.



Geometry and tracking

8

Example of cell unfolding in direction X :

6	5	5	6	6	5	5	6
4	(3 1) 2	2 (1 3)	4	4	(3 1) 2	2 (1 3)	4

5	6	5	6	5	6	5	6
2 (1 3)	4	2 (1 3)	4	2 (1 3)	4	2 (1 3)	4



Geometry and tracking

9

Simplifying CP equations using approximate BC:

- Assume:

$$\vec{J}_{-}^g = \mathbf{A}^g \vec{J}_{+}^g$$

- Final transport equation:

$$\vec{\phi}^g = \mathbf{P}_{c,vv}^g \vec{q}^g$$

with the complete collision probability matrix $\mathbf{P}_{c,vv}^g$:

$$\mathbf{P}_{c,vv}^g = \left(\mathbf{P}_{vv}^g + \mathbf{P}_{vs}^g ((\mathbf{A}^g)^{-1} - \mathbf{P}_{ss}^g)^{-1} \mathbf{P}_{sv}^g \right)$$



Geometry and tracking

10

Numerical CP evaluation:

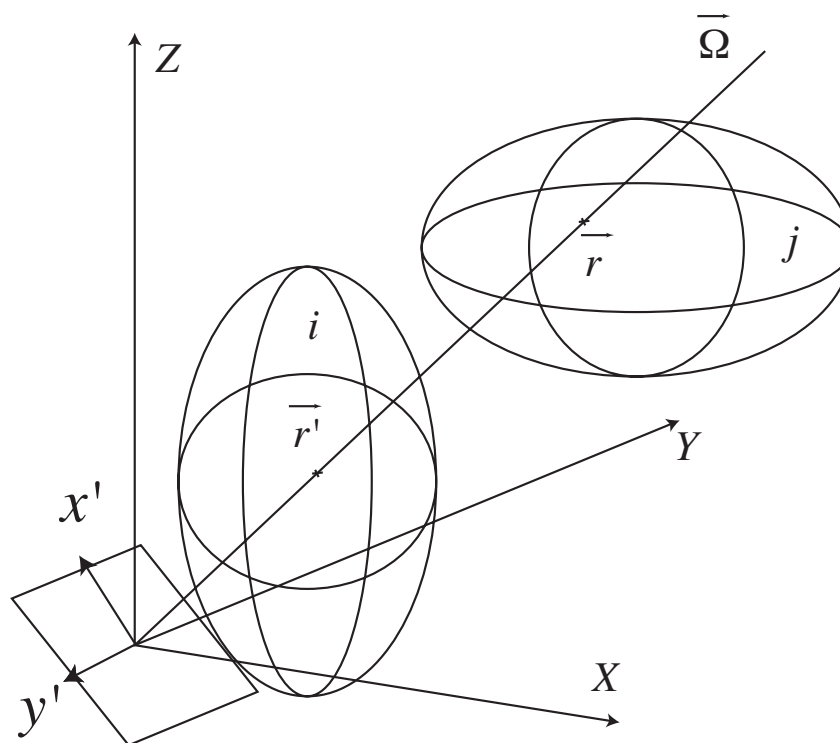
$$\begin{aligned} \tilde{p}_{ij}^g = & \int dx' \int dy' \int dR' \int_{4\pi} d^2\Omega \int_{R_i - \frac{1}{2}}^{R_i + \frac{1}{2}} dR \\ & \times \frac{e^{-\tau^g(R)}}{4\pi} \Theta_i \Theta_j \end{aligned}$$



Geometry and tracking

11

General 3-D geometry for collision probability integration:

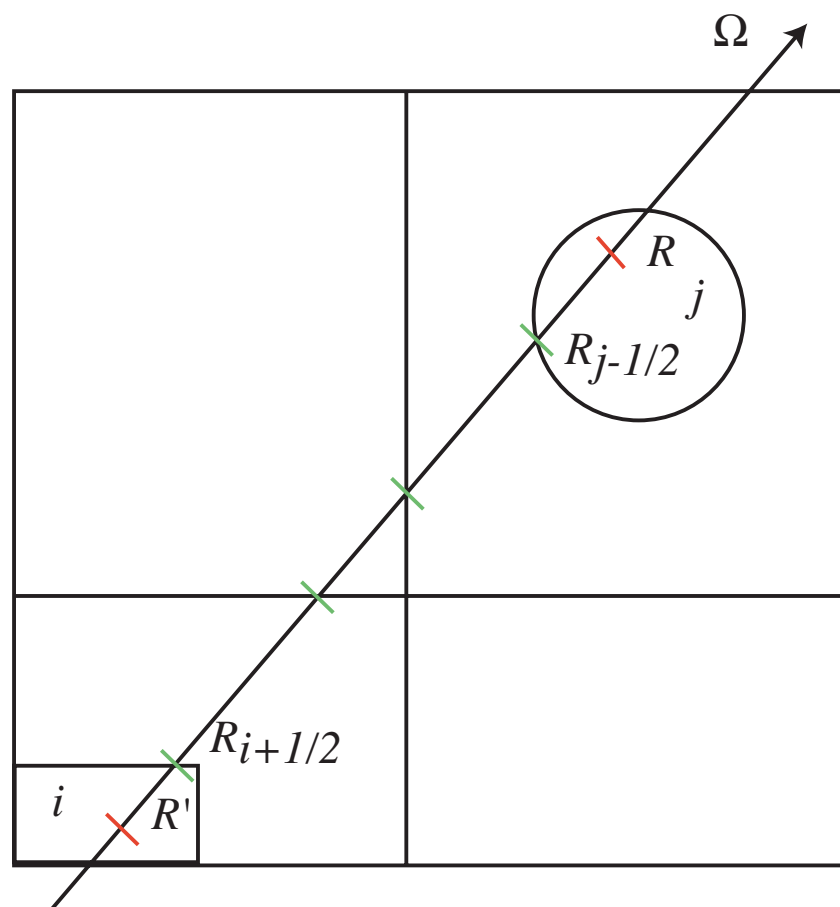




Geometry and tracking

12

Projection of 3–D geometry on a 2–D plane:





Geometry and tracking

13

After integration over R' and R , one obtains:

$$\begin{aligned}\tilde{p}_{ij}^g &= \frac{1}{4\pi \Sigma_i^g \Sigma_j^g} \int_0^{4\pi} d^2\Omega \int dx' \int dy' \\ &\times \left[1 - \exp\left(-\tau_{i-\frac{1}{2}, i+\frac{1}{2}}^g\right) \right] \exp\left(-\tau_{i+\frac{1}{2}, j-\frac{1}{2}}^g\right) \\ &\times \left[1 - \exp\left(-\tau_{j-\frac{1}{2}, j+\frac{1}{2}}^g\right) \right] \Theta_i \Theta_j\end{aligned}$$

with:

$$\begin{aligned}\tau_{i\pm\frac{1}{2}, j\pm\frac{1}{2}}^g &= \Sigma_i^g (R_{i+\frac{1}{2}} - R_{i\pm\frac{1}{2}}) + \tau_{i+\frac{1}{2}, j-\frac{1}{2}} \\ &\quad + \Sigma_j^g (R_{j\pm\frac{1}{2}} - R_{j-\frac{1}{2}})\end{aligned}$$



Geometry and tracking

14

- For 3–D geometry, the final integrations are evaluated numerically:

$$\begin{aligned}\tilde{p}_{ij}^g = & \frac{1}{4\pi \Sigma_i^g \Sigma_j^g} \sum_{\alpha=1}^{N_\Omega} W_\Omega \sum_{m=1}^{N_\rho} \sum_{n=1}^{N_\rho} \delta^2 \\ & \times \left[1 - \exp \left(-\tau_{i-\frac{1}{2}, i+\frac{1}{2}}^g \right) \right] \exp \left(-\tau_{i+\frac{1}{2}, j-\frac{1}{2}}^g \right) \\ & \times \left[1 - \exp \left(-\tau_{j-\frac{1}{2}, j+\frac{1}{2}}^g \right) \right] \Theta_i \Theta_j\end{aligned}$$

and a 4–D numerical quadrature remains.

- For 2–D geometry the integration over the polar angle leads to Bickley-Naylor functions and a 2–D numerical quadrature remains.



Geometry and tracking

15

For the method of characteristics the flux is evaluated using:

$$\begin{aligned}
 V_i \phi_i = & \sum_{\alpha=1}^{N_{\Omega}} W_{\Omega} \sum_{m=1}^{N_{\rho}} \sum_{n=1}^{N_{\rho}} \delta^2 \sum_{j=1}^{i-1} q_j \left(\frac{1 - e^{-\tau_j(\vec{\rho}_{\alpha,m,n})}}{\Sigma_j} \right) \\
 & \left[e^{-\sum_{l=j+1}^{i-1} \tau_l(\vec{\rho}_{\alpha,m,n})} \right] \left(\frac{1 - e^{-\tau_i(\vec{\rho}_{\alpha,m,n})}}{\Sigma_i} \right) \\
 & + \sum_{\alpha=1}^{N_{\Omega}} W_{\Omega} \sum_{m=1}^{N_{\rho}} \sum_{n=1}^{N_{\rho}} \delta^2 q_i \left(\frac{\tau_i(\vec{\rho}_{\alpha,m,n}) - 1 + e^{-\tau_i(\vec{\rho}_{\alpha,m,n})}}{\Sigma_i^2} \right)
 \end{aligned}$$

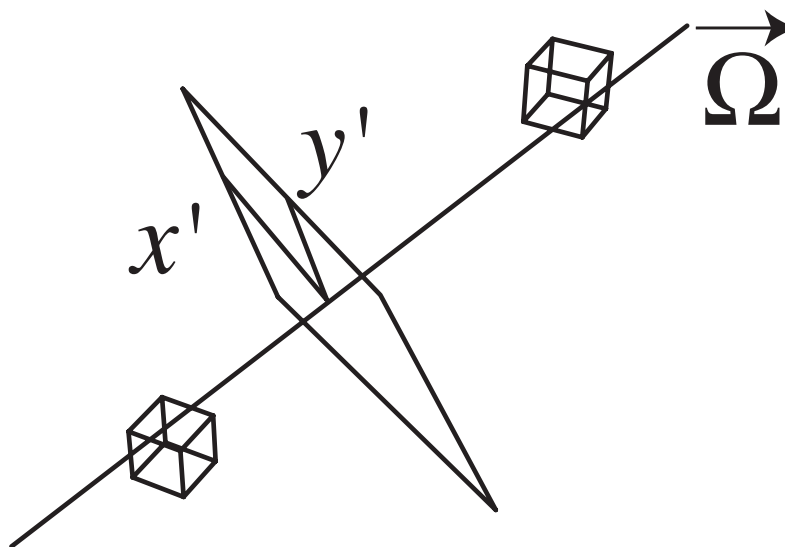
The numerical quadrature used is identical to that considered for the CP method.



Geometry and tracking

16

Quadrature for Cartesian 3-D geometry:





Geometry and tracking

17

Angular quadrature for 3-D Cartesian geometry:

- S_n type EQ_N quadrature with $4N_\Omega(N_\Omega + 2)/8$ angular directions $\vec{\Omega}_{1,i}$, $\vec{\Omega}_{2,i}$, $\vec{\Omega}_{3,i}$ and $\vec{\Omega}_{4,i}$.
- Global quadrature weight $W_\Omega = 2/(N_\Omega(N_\Omega + 2))$.
- Tracking in upper half sphere only.

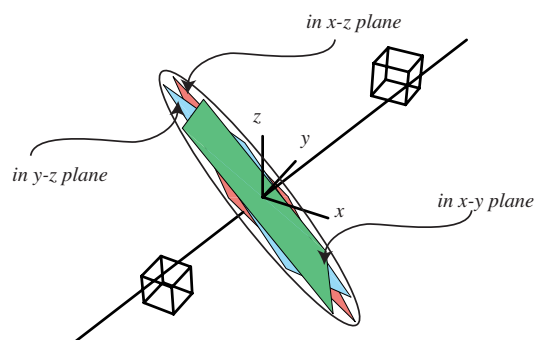


Geometry and tracking

18

Comments on Cartesian surface quadrature in 3-D:

- 3-D different planes selected for each direction.





Geometry and tracking

19

Spatial quadrature for Cartesian 3–D geometry:

- Locate the (x'_c, y'_c) center of the geometry.
- Compute radius h_+ of smallest sphere containing the geometry.
- From track density ρ compute:

$$N_\rho = (2\sqrt{\rho}h_+) + 1 \qquad \delta = \frac{2h_+}{N_\rho}$$

- Integration line l_{mn} passes through:

$$x'_m = x'_c + \left(\frac{2m-1}{2}\right)\delta \qquad y'_n = y'_c - h_+ + \left(\frac{2n-1}{2}\right)\delta$$



Geometry and tracking

20

Angular quadrature for 2-D Cartesian geometry:

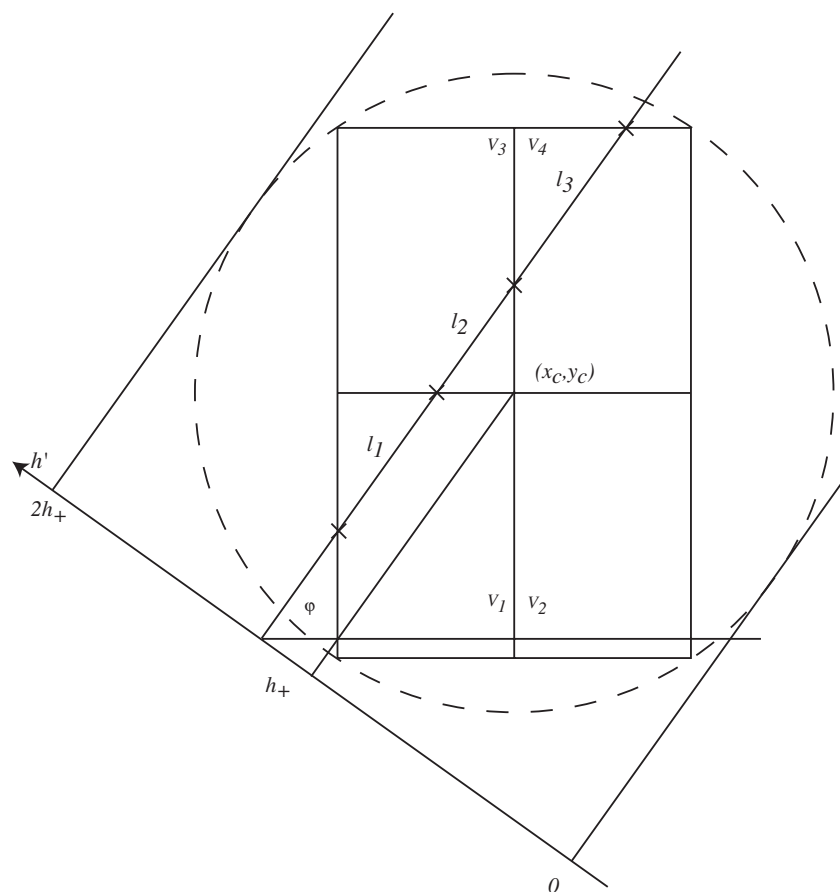
- Order N trapezoidal quadrature for azimuth φ .
- Tracking for $0 \leq \varphi \leq \pi$.
- For standard tracking, use Bickley-Nayler functions for polar (θ) quadrature.
- For cyclic tracking use order $(N + 1)/2$ Gauss-Legendre quadrature for polar (θ) angle.



Geometry and tracking

21

CP quadrature for Cartesian 2-D geometry:





Geometry and tracking

22

Angular quadrature for 3-D Hexagonal geometry:

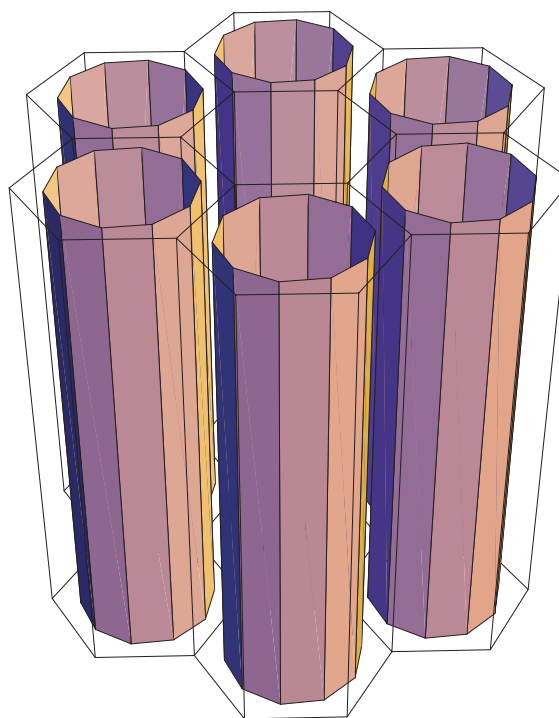
- Order N trapezoidal quadrature for azimuth φ .
- Order $(N + 1)/2$ Gauss-Legendre quadrature for polar angle.
- Tracking in upper half sphere only.



Geometry and tracking

23

Example of valid hexagonal 3-D geometry:

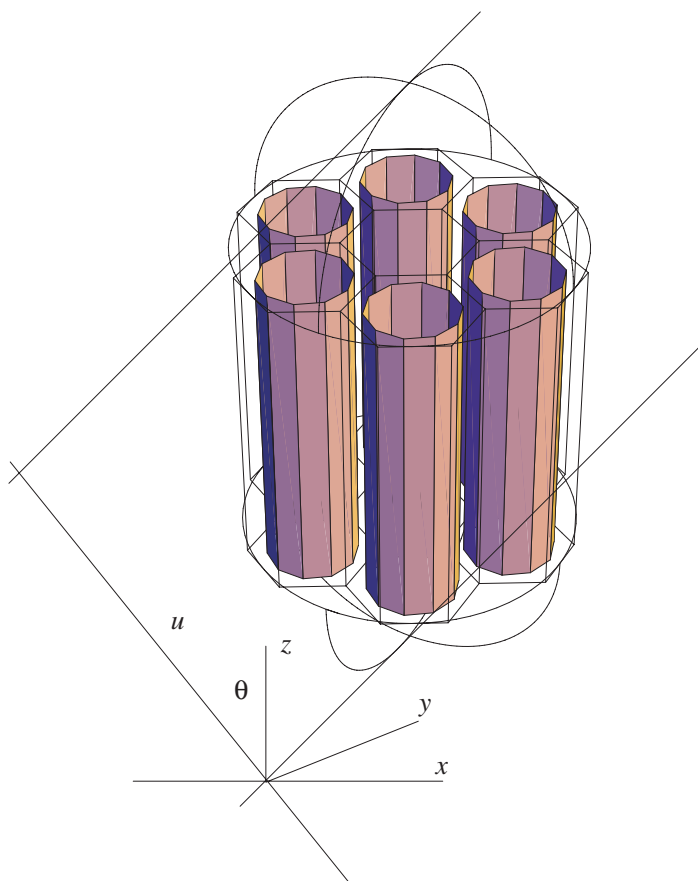




Geometry and tracking

24

Angular quadrature for hexagonal 3-D geometry:





Geometry and tracking

25

Angular quadrature for 2-D Hexagonal geometry:

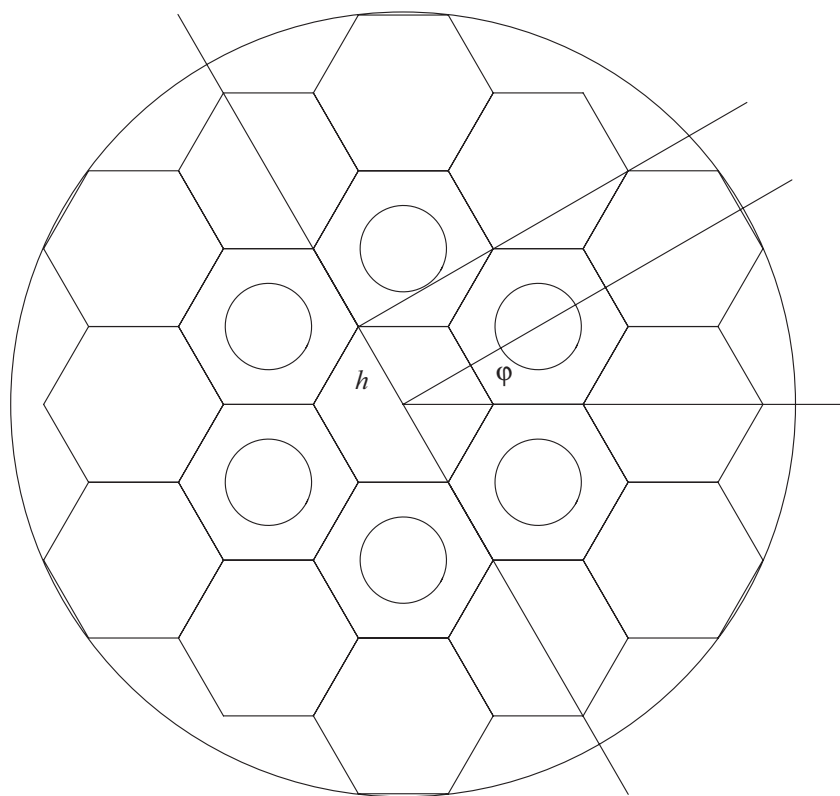
- Order N trapezoidal quadrature for azimuth.
- Tracking for $0 \leq \varphi \leq \pi$.
- Use Bickley-Nayler functions for polar quadrature.



Geometry and tracking

26

Angular quadrature for hexagonal 2-D geometry:





Geometry and tracking

27

Comments on angular quadrature:

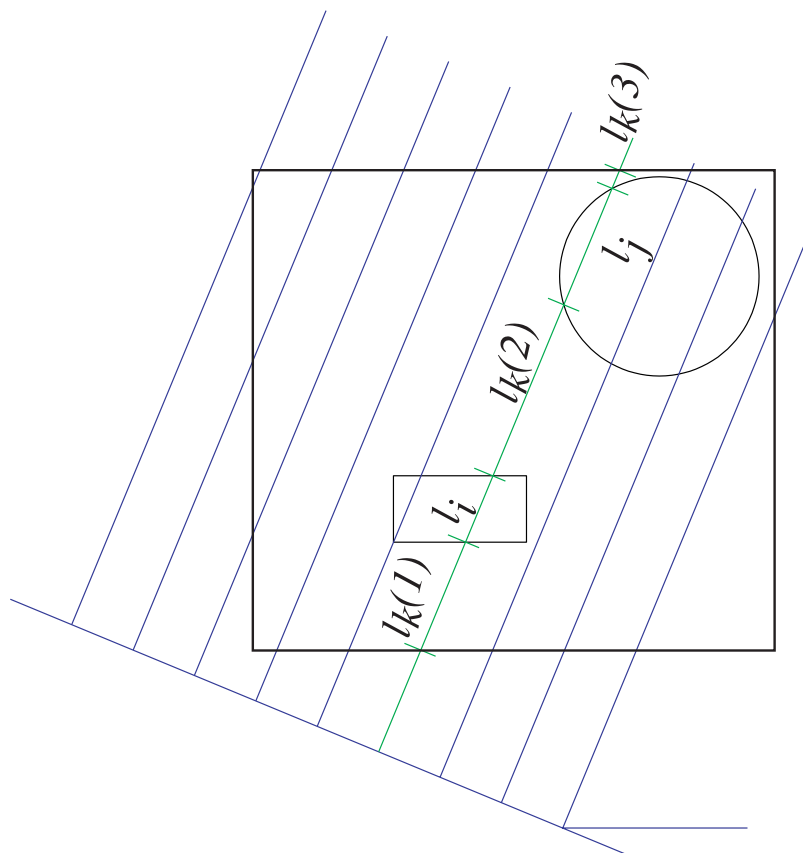
- Tracking in the lower half sphere (3–D) or circle (2–D) is not considered because of symmetry of CP matrices.
- Try to select as many angles as possible (neutrons travel on a straight line).
- Try to select as high a density as possible (to maximize the number of lines crossing each region).
- For 3–D Cartesian cases, the angular quadrature is currently limited to $N_{\Omega} = 16$ (144 angles).



Geometry and tracking

28

DRAGON tracking example:





Geometry and tracking

29

DRAGON tracking of a line $l_{m,n}$:

1. Follow the tracking line as it travels through the cell.
2. Identify each region i and surface s uniquely.
3. Compute distance $\tilde{l}_{i,m,n}$ the neutron travels in each region.
4. Associate region flux number I with set of spatial regions i .
5. Associate surface flux number S with sets of surfaces s .
6. Store information on temporary tracking file.



Geometry and tracking

30

Post treatment of tracking file:

- For each direction, normalize tracks using:

$$l_{i,m,n} = \left(\frac{V_i}{\tilde{V}_i} \right) \tilde{l}_{i,m,n}$$

where:

$$\tilde{V}_i = \frac{W_p}{3} \sum_{m=1}^{N_p} \sum_{n=1}^{N_p} \tilde{l}_{i,m,n}$$



Geometry and tracking

31

Comments on storage requirements for tracking:

- Maximum number of line segments tracks d_t :

$$d_t < 6N(N_\Omega(N_\Omega + 2))\rho_p h_+^2$$

- For $N = 1000$, $h_+ = 50$, $\rho_p = 20 \text{ t/cm}^2$ and $N_\Omega = 8$:

$$d_t < 20 \times 10^9$$

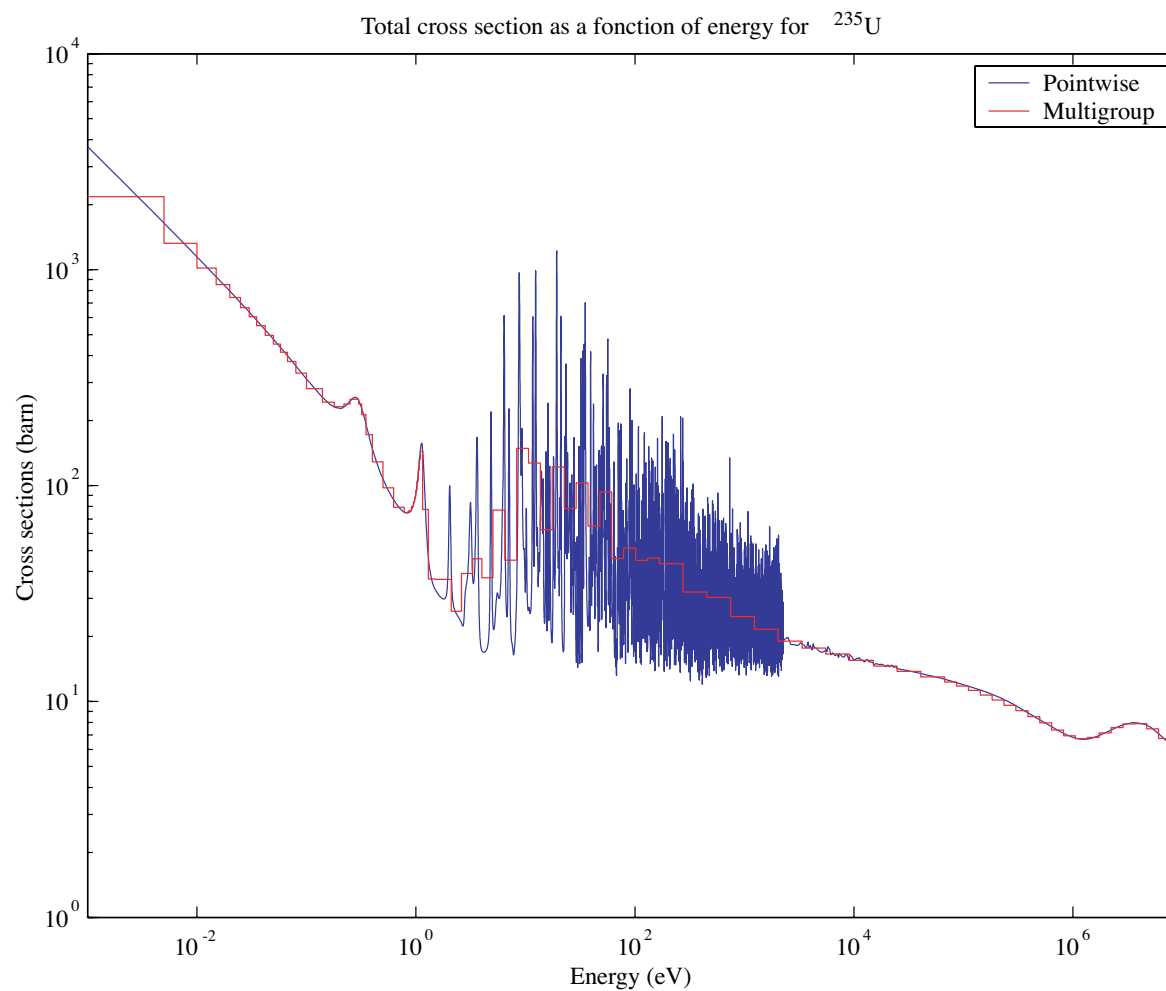
tracks segments.



Self-Shielding

1

Multigroup cross section versus continuous energy:

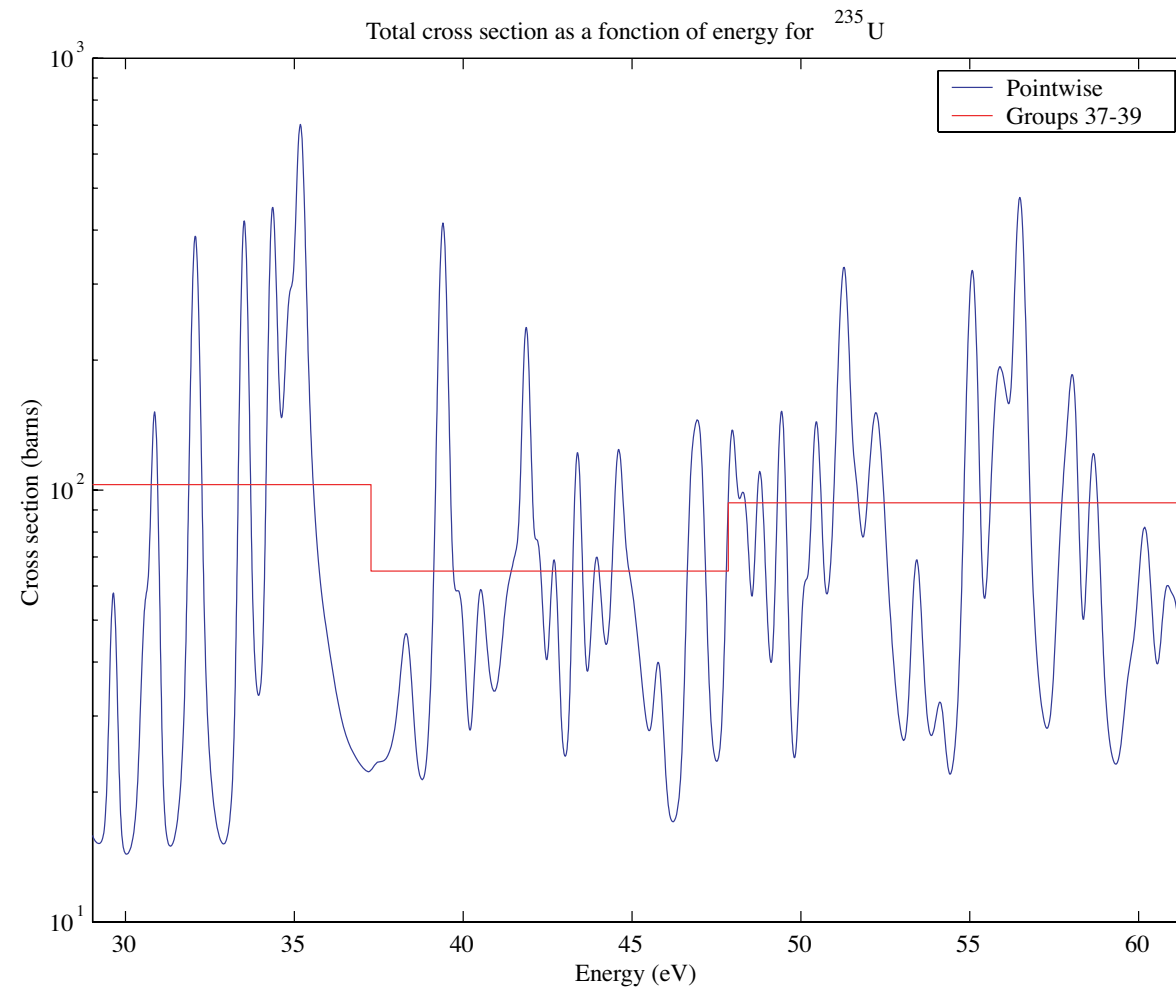




Self-Shielding

2

Multigroup cross section versus continuous energy:





Self-Shielding

3

Multigroup condensation procedure is selected to preserve the reaction rates (resonance integrals):

$$R_{I,x}^g(\sigma_0) = \int_{E_g}^{E_{g-1}} \sigma_{I,x}(E) \phi(E, \sigma_0)$$

Here, an approximated flux distribution $\phi(E, \sigma_0)$ is used corresponding to:

- An Infinite cell calculation for a binary mixture of hydrogen and isotope I .

The resonance integrals are tabulated as a function of the dilution σ_0 cross section and stored in the microscopic cross section library.



Self-Shielding

4

Multigroup cross sections are computed using:

$$\sigma_{I,x}^g(\sigma_e) = \frac{R_{I,x}^g(\sigma_e)}{\phi^g(\sigma_e)}$$

Here:

- The effective dilution σ_e is approximated using the self-shielding process.
- $R_{I,x}^g(\sigma_e)$ are extracted from the library and interpolated in dilution.
- $\phi^g(\sigma_e)$ can be extracted from library of computed approximately.



Self-Shielding

5

Resonance self-shielding:

- Required to evaluate the equivalent dilution cross section σ_e at which $R_{I,x}^g$ will be evaluated.
- Dilution cross section will be dependent on space, and mixture considered.
 - Cell is finite and heterogeneous.
 - Mixture not necessarily a binary mixture of hydrogen and isotope I .
- Use equivalence principle:
 - Two systems with the same value of σ_e will have the same self-shielded cross section.



Self-Shielding

6

Improved Stamm'ler method:

1. Compute fuel-to-fuel collision probability $P_{\text{ff}}(\sigma_e)$ in exact geometry for arbitrary dilutions σ_e .
2. Approximate $P_{\text{ff}}(\sigma_e)$ by a rational series in σ_e and invert series $[\sigma_e(P_{\text{ff}})]$.
3. Start iteration by using $\tilde{\sigma}_e$ an initial dilution.
4. Evaluate self-shielded cross sections $\sigma_{I,x}^g(\tilde{\sigma}_e)$ and compute exact P_{ff} .
5. Evaluate new approximation $\tilde{\sigma}_e$ using rational series.
6. Check for convergence and exit if convergence reached otherwise return to 4.



Self-Shielding

7

Self-shielded flux evaluation:

- Interpolate in table if $\phi^g(\sigma_0)$ is tabulated as a function of dilution.
- Approximate using the conservation relation in an homogeneous media:

$$\phi^g(\sigma_e) = 1 - \frac{1}{N\sigma_e} \left[R_I^g(\sigma_e) - \sum_h \frac{U^h}{U^g} R_{I,s}^{h \rightarrow g}(\sigma_e) \right]$$



Self-Shielding

8

- Approximate using the conservation relation in an heterogeneous media:

$$\phi^g(\sigma_e) = 1 - \frac{1}{E(\sigma_e)} \left[R_I^g(\sigma_e) - \sum_h \frac{U^h}{U^g} R_{I,s}^{h \rightarrow g}(\sigma_e) \right]$$

where $E(\sigma_e)$ is the escape matrix defined as:

$$E(\sigma_e) = \frac{1}{P_{\text{ff}}(\sigma_e)} - N\sigma_{I,x}^g(\sigma_e)$$

This last approximation is known as the Livolant-Jeanpierre normalization.



Homogenization/Condensation

1

The DRAGON editing module EDI : performs the following tasks:

- Condense the multigroup microscopic and macroscopic cross sections.
- Homogenize the multigroup microscopic and macroscopic cross sections.
- Prints the reaction rates and the homogenize/condense cross sections.
- Save this information for further used in the form of a MACROLIB or a MICROLIB.



Homogenization/Condensation

2

Condensation and homogenization based on the following assumptions:

- Reaction rates are physically meaningful and should be preserved by the condensation/homogenization procedure.

$$R_i = \sum_g V_i \phi_i^g \Sigma_i^g = V_i \phi_i \Sigma_i$$

$$R^g = \sum_i V_i \phi_i^g \Sigma_i^g = V \phi^g \Sigma^g$$

- The eigenvalue is physically meaningful and should be preserved by the condensation/homogenization procedure.



Homogenization/Condensation

3

Condensation:

- Condensed transport equation (macrogroup K):

$$\vec{\Omega} \cdot \vec{\nabla} \sum_{g \in G_K} \Phi^g(\vec{r}, \vec{\Omega}) + \sum_{g \in G_K} \Sigma^g(\vec{r}) \Phi^g(\vec{r}, \vec{\Omega}) = \sum_{g \in G_K} [Q_s^g(\vec{r}, \vec{\Omega}) + \frac{1}{k} Q_f^g(\vec{r})]$$

- Few group version of the same equation:

$$\vec{\Omega} \cdot \vec{\nabla} \Phi^K(\vec{r}, \vec{\Omega}) + \Sigma^K(\vec{r}) \Phi^K(\vec{r}, \vec{\Omega}) = [Q_s^K(\vec{r}, \vec{\Omega}) + \frac{1}{k} Q_f^K(\vec{r}, \vec{\Omega})]$$

It should reproduce condensed multigroup results.



Homogenization/Condensation

4

- The condensation procedure that satisfies our requirements:

$$\phi_i^K = \sum_{g \in G_K} \phi_i^g$$

$$\Sigma_i^K = \frac{1}{\phi_i^K} \sum_{g \in G_K} \Sigma_i^g \phi_i^g$$

$$\Sigma_{s,i}^{L \rightarrow K} = \frac{1}{\phi_i^L} \sum_{h \in G_L} \sum_{g \in G_K} \Sigma_{s,i}^{h \rightarrow g} \phi_i^h$$

$$\chi_i^K = \sum_{g \in G_K} \chi_i^g$$

$$\nu \Sigma_{f,i}^K = \frac{1}{\phi_i^K} \sum_{g \in G_K} \nu \Sigma_{f,i}^h \phi_i^g$$



Homogenization/Condensation

5

Full cell homogenization:

- Homogenized transport equation (no leakage):

$$\sum_i \Sigma_i^g V_i \phi_i^g = \sum_i V_i [Q_{s,i}^g + \frac{1}{k} Q_{f,i}^g]$$

- Homogeneous transport equation (no leakage):

$$\hat{\Sigma}^g V \hat{\phi}^g = V [\hat{Q}_{s,i}^g + \frac{1}{k} \hat{Q}_{f,i}^g]$$

- Homogenized and homogeneous transport equations are identical if one selects a flux-volume homogenization technique.



Homogenization/Condensation

6

Flux-volume homogenization technique:

$$\hat{\phi}^g = \frac{1}{V} \sum_i V_i \phi_i^g$$

$$\hat{\Sigma}^g = \frac{1}{V \hat{\phi}^g} \sum_i V_i \Sigma_i^g \phi_i^g$$

$$\hat{\Sigma}_s^{h \rightarrow g} = \frac{1}{V \hat{\phi}^h} \sum_i V_i \Sigma_{s,i}^{h \rightarrow g} \phi_i^h$$

$$\hat{\nu} \hat{\Sigma}_f^g = \frac{1}{V \hat{\phi}^g} \sum_i V_i \nu \Sigma_{f,i}^g \phi_i^g$$

$$\hat{\chi}^g = \frac{1}{V \sum_h \hat{\nu} \hat{\Sigma}_f^g \hat{\phi}^g} \sum_i \chi_i^g V_i \sum_h \nu \Sigma_{f,i}^g \phi_i^h$$



Homogenization/Condensation

7

Flux-volume homogenization fails if:

- The cell is finite (a cell with leakage):

$$\sum_{j=1}^{N_j} p_{ij}^g \Sigma_j^g \neq 1$$

- Partial cell homogenization is considered.



Homogenization/Condensation

8

Partial homogenization:

- The N region transport equation homogenized over M regions:

$$\sum_{i \in M_I} V_i \Sigma_i^g \phi_i^g = \sum_{i \in M_I} \sum_J \sum_{j \in M_J} p_{ji}^g(\Sigma^g) [Q_{s,i}^g + \frac{1}{k} Q_{f,i}^g]$$

- The M region transport equation:

$$V_I \hat{\Sigma}_I^g \hat{\phi}_I^g = \sum_J \hat{p}_{JI}^g(\hat{\Sigma}^g) [Q_{s,J}^g + \frac{1}{k} Q_{f,J}^g]$$

- $\hat{P}_{JI}^g(\hat{\Sigma}^g)$ are CP computed using homogenized cross sections.



Homogenization/Condensation

9

We need:

$$\sum_{i \in M_I} V_i \Sigma_i^g \phi_i^g = V_I \Sigma_I^g \phi_I^g$$

and:

$$\begin{aligned} \sum_J \hat{p}_{JI}^g(\hat{\Sigma}^g) [Q_{s,J}^g + \frac{1}{k} Q_{f,J}^g] = \\ \sum_{i \in M_I} \sum_J \sum_{j \in M_J} p_{ji}^g(\Sigma^g) [Q_{s,i}^g + \frac{1}{k} Q_{f,i}^g] \end{aligned}$$

to be simultaneously true.



Homogenization/Condensation

10

The flux-volume homogenization method is not longer adequate because:

- There is no simple relation between $\hat{p}_{JI}^g(\hat{\Sigma}^g)$ and $p_{ji}^g(\Sigma^g)$.

The alternative here is to use a non-linear process:

- Consider a flux-volume homogenization for ϕ_I^g and Σ_I^g .
- Redefine the homogeneous flux $\hat{\phi}_I^g$ and cross sections $\hat{\Sigma}_I^g$ as follows:

$$\hat{\phi}_I^g = \frac{1}{\mu_I^g} \phi_I^g \quad \hat{\Sigma}_I^g = \mu_I^g \Sigma_I^g$$



Homogenization/Condensation

11

- Determine the SPH factors μ_I^g such a way that:

$$\sum_J \hat{p}_{JI}^g(\hat{\Sigma}^g)[Q_{s,J}^g + \frac{1}{k}Q_{f,J}^g] =$$

$$\sum_{i \in M_I} \sum_J \sum_{j \in M_J} p_{ji}^g(\Sigma^g)[Q_{s,i}^g + \frac{1}{k}Q_{f,i}^g]$$

is true.

- This definition ensures that:

$$\sum_{i \in M_I} V_i \Sigma_i^g \phi_i^g = V_I \Sigma_I^g \phi_I^g = V_I \tilde{\Sigma}_I^g \tilde{\phi}_I^g$$



Homogenization/Condensation

12

Microscopic cross section:

- The macroscopic cross section associated with region m is:

$$\Sigma_m^g = \sum_I \Sigma_{I,m}^g$$

- $\Sigma_{I,m}^g$ is the contribution of isotope I to Σ_m^g :

$$\Sigma_{I,m}^g = N_{I,m} \sigma_I^g$$

- $N_{I,m}$, the concentration of isotope I in region m .
- The homogenization and condensation procedure described above remain valid for $\Sigma_{I,m}^g$.



Homogenization/Condensation 13

- The concentration of isotope I in homogenized region M is:

$$N_{I,M} = \frac{1}{V_M} \sum_{m \in M} N_{I,m} V_m$$

- The equivalent homogenized microscopic cross section is therefore:

$$\hat{\sigma}_{I,M}^g = \frac{\mu_M^g}{N_{I,M} V_M \phi_M^g} \sum_{m \in M} N_{I,m} V_m \sigma_I^g \phi_m^g$$

- The microscopic cross sections now become dependent on the spatial position.



Isotopic Depletion

1

The isotopic depletion equations:

$$\frac{dN_I(t)}{dt} = \sum_J S_I^J(t) - N_I(t) \left(\int dE \sigma_{I,a}(E) \phi(E, t) + \lambda_I \right)$$

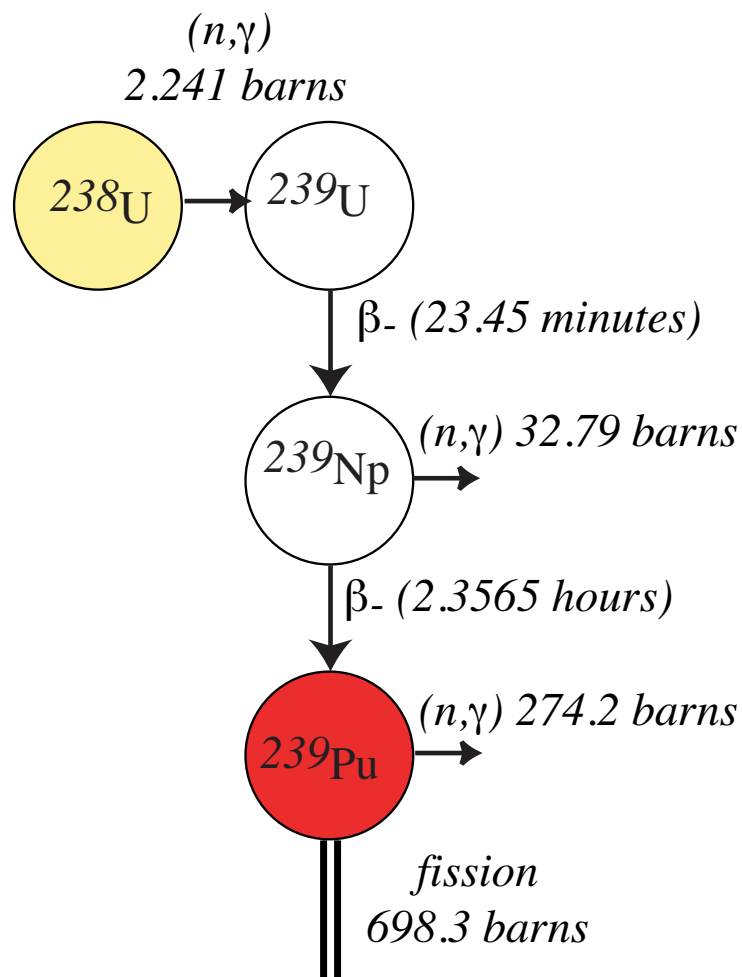
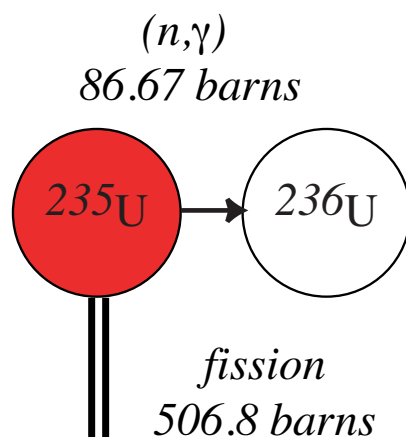
- λ_I is the decay rate of I .
- $\sigma_{I,a}(E)$ is the neutron absorption cross section for I .
- $S_I(t)$ is the production rate for isotope I from isotope J .
- Initial concentrations $N_I(t_0)$ required.



Isotopic Depletion

2

Example of simplified decay chains for ^{235}U and ^{238}U .



● Fertile isotope
● Fissile isotope



Isotopic Depletion

3

Approaches used in DRAGON to solve the isotopic depletion equations:

- Use a quasi-static approach assuming the flux vary slowly over macro time steps (linear variation).
- Solve depletion equation using a Runge-Kutta or Kaps-Renthrop algorithm inside macro time steps with linear flux variations.



Isotopic Depletion

4

Flux normalization options available:

- Constant flux burnup.
- Constant power burnup.
- Out of core depletion.



Isotopic Depletion

5

Fuel depletion calculations:

- Depletion is controlled using the time variable.
- Selection of depletion time step is important.
- Includes an isotope saturation model.

For out of core depletion calculations:

- A single time step is sufficient since flux vanishes.

For in core depletion calculations:

- Start with short Δt .
- Increase Δt using a log scale to follow spatial and energy variation in flux as a function of time.



Isotopic Depletion

6

- Burnup at fixed power uses an inner iterative process (inside EVO:).
 - Energy and spatial flux distribution are assumed constant.
 - Assume integrated flux level has a linear time dependence.
 - Iterate internally between final flux level and isotopic concentrations to ensure coherence between initial and final power production.



Isotopic Depletion

7

For in core depletion calculations:

- Burnup at fixed power and flux levels can also use an outer iterative process.
- When energy and spatial flux distribution change are strongly affected by depletion.
- Iterate externally between flux calculations (FLU:) and isotopic concentrations (EVO:).
- This ensures that the final isotopic concentrations are coherent with the final energy and spatial flux distribution.



Status of DRAGON

G. Marleau

Institut de génie nucléaire
École Polytechnique de Montréal



Status of DRAGON

1

Contents:

1. Verification.
2. Validation.
3. Documentation.
4. Using DRAGON.



Verification

1

DRAGON verification exercise:

1. The EXCELT: Module.
2. The ASM: Module.
3. The FLU: Module.
4. The EDI: Module



Verification

2

DRAGON initial verification:

- Select the DRAGON modules that are critical to 3-D CANDU applications.
- Select a series of routines in these modules for line by line verification.
- Select routines or blocks of routines for numerical verification.
 - Self-consistency tests.
 - Verification using an independent method or program.

Here I will concentrate on the numerical verification.



Verification

3

The EXCELT: Module:

- To analyze an already defined geometry.
- To assign to each region and surface an index and to compute the volume and surface area.
- Using quadrature parameters, to generate integration lines.
- Verify the integration lines and perform normalization if required.

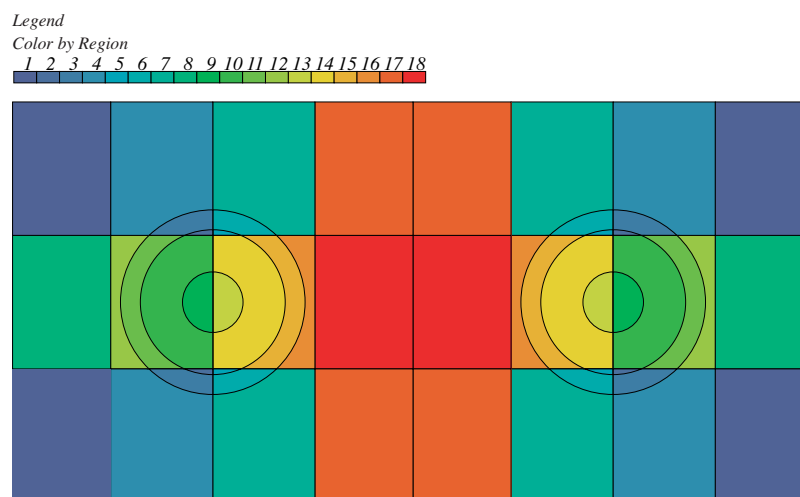


Verification

4

The TRACKING data structure also contains a description of the geometry:

- Compare contents of TRACKING and GEOMETRY data structure.
- Use PSP: module to illustrate 2-D geometry:

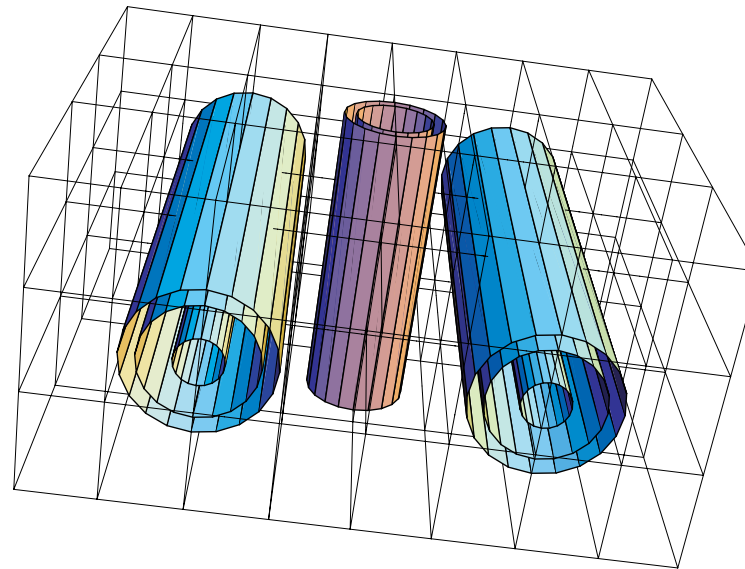




Verification

5

- Use Mathematica to illustrate 2-D and 3-D geometries using the information provided on the tracking data structure:



- A Mathematica interface was written to access the TRACKING data structure.

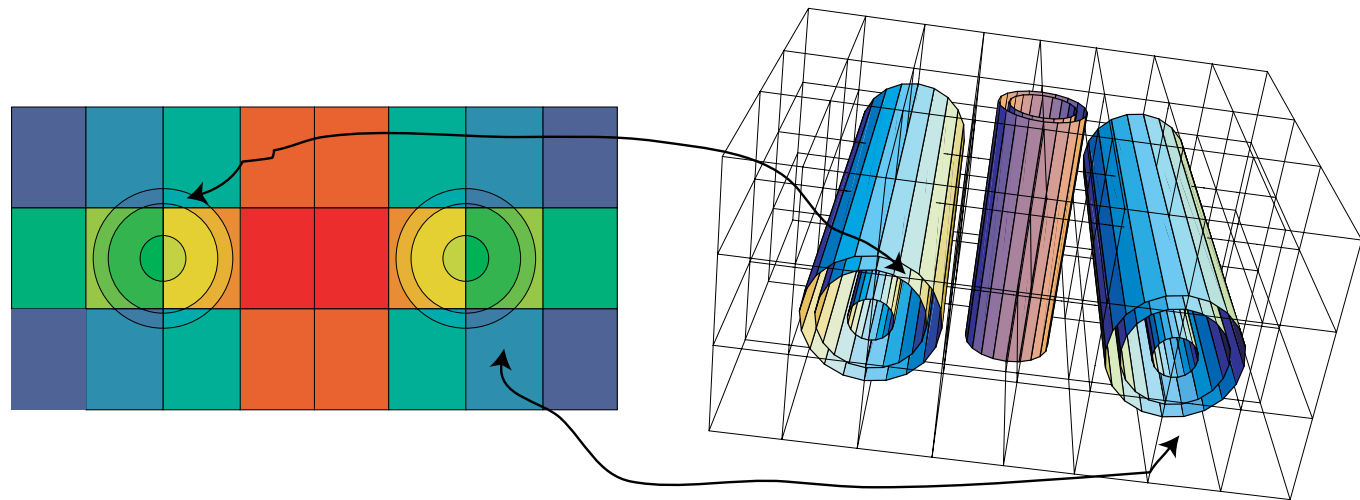


Verification

6

Surface area and region volume evaluation:

- Comparison of DRAGON region volumes with explicit calculations.
- Comparison of DRAGON volume for equivalent 2-D and 3-D geometries:





Verification

7

Binary tracking file:

- Numerical evaluation of volume and comparison with exact values:

$$V_i = h \sum_k l_{k,i}$$

- h is spatial integration weight.
- $l_{k,i}$ is path length of line k in region i .



Verification

8

2–D tracking volume errors:

Integration parameters			Errors (%)	
Angular	Spatial	Nb. lines	Maximum	Average
3	1.0	163	10.5	0.45
3	10.0	1637	1.1	0.03
3	100.0	16376	0.1	0.01
31	1.0	1671	11.0	0.03
31	10.0	16917	0.21	0.00
31	100.0	169167	0.01	0.00
91	1.0	4907	9.3	0.01
91	10.0	49673	0.19	0.00
91	100.0	496581	0.01	0.00



Verification

9

3–D tracking volume errors:

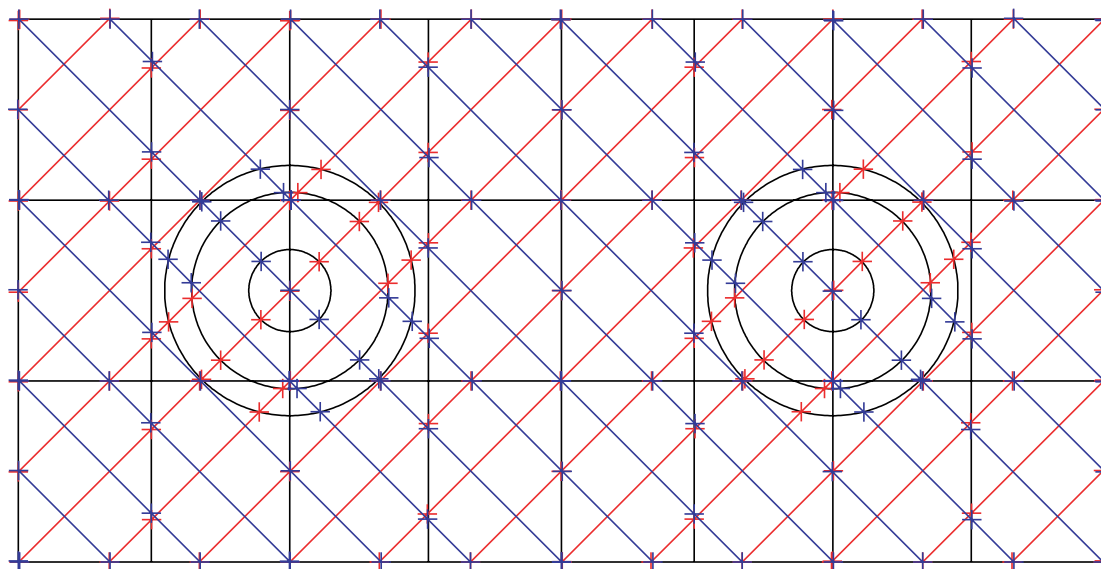
Integration parameters			Errors (%)	
Angular	Spatial	Nb. lines	Maximum	Average
2	1.0	40820	6.61	-0.003
2	5.0	203660	1.49	0.001
2	25.0	1021544	0.29	0.001
4	1.0	110644	6.61	-0.000
4	5.0	555118	1.49	-0.000
4	25.0	2775748	0.20	0.000
8	1.0	359432	5.75	0.003
8	5.0	1793640	0.98	0.001
8	25.0	8982529	0.37	0.000



Verification

10

Extract integration line from binary tracking fine and plot over geometry using Matlab in 2-D:

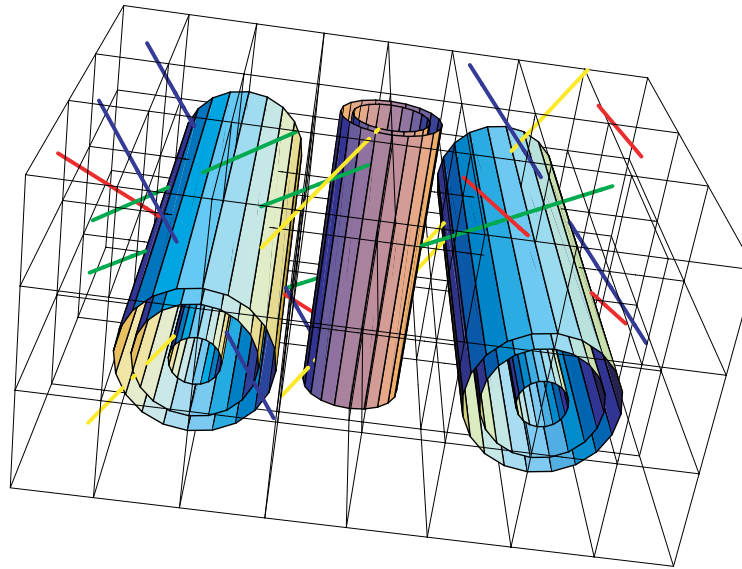




Verification

11

Extract integration line from binary tracking fine and plot over geometry using Mathematica in 3-D:



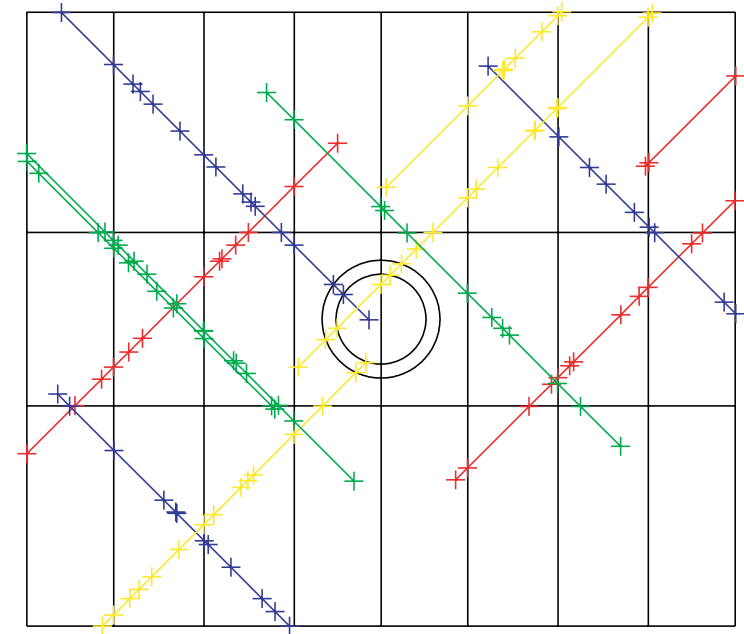
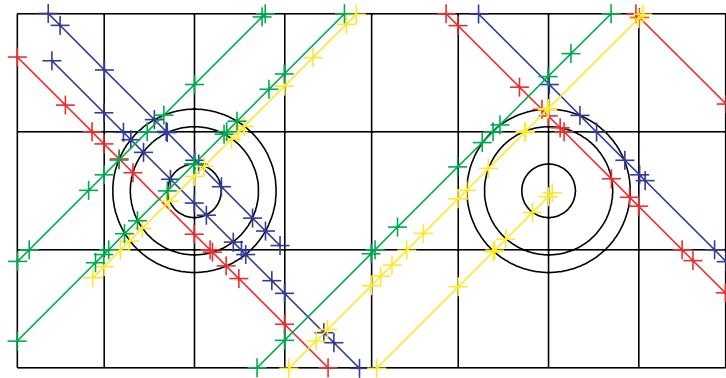
- 12540 lines generated.
- Only a few lines plotted.
- 16/178 surfaces not touched by lines.



Verification

12

2-D projection of 3-D tracking lines using Matlab:





Verification

13

The ASM: Module:

- To compute the collision, leakage and escape probability matrix using a set of integration lines.
- To verify the CP conservation laws and perform CP normalization if required.
- To take into account the boundary conditions (approximately or exactly).
- To take into account within-group scattering.



CP conservation relations:

$$\frac{4}{S_\beta} \left(\sum_{\alpha=1}^{N_\alpha} \frac{S_\alpha}{4} p_{\alpha\beta} + \sum_{i=1}^{N_i} \Sigma_i V_i p_{i\beta} \right) = R_\beta$$

$$\frac{1}{V_j} \left(\sum_{\alpha=1}^{N_\alpha} \frac{S_\alpha}{4} p_{\alpha j} + \sum_{i=1}^{N_i} \Sigma_i V_i p_{ij} \right) = R_j$$

- If conservation relations satisfied: $R_\beta = R_j = 1$.
- Error on conservation relations:

$$\epsilon_{\beta/j} = 100 \times \left(\frac{||1 - R_{\beta/j}||}{||R_{\beta/j}||} \right)$$



Verification

15

Error on conservation relations:

- Computed directly in DRAGON.
- Before CP normalization $\epsilon_{\beta/j}^B$.
- After CP normalization $\epsilon_{\beta/j}^A$.

Parameter	T2D	T3D
ϵ_{β}^B	0.23	4.36
ϵ_j^B	0.000003	0.000008
ϵ_{β}^A	0.00004	0.00002
ϵ_j^A	0.00003	0.00003



Verification

16

- Complete CP matrix:

$$\mathbf{P}_{c,vv}^g = (\mathbf{P}_{vv}^g + \mathbf{P}_{vs}^g (\mathbf{A}^{-1} - \mathbf{P}_{ss}^g)^{-1} \mathbf{P}_{sv}^g)$$

where \mathbf{A} is the transmission/reflection matrix.

- Scattering modified CP matrix:

$$\mathbf{W}_{vv}^g = \left(\mathbf{I} - \mathbf{P}_{c,vv}^g \Sigma_{s,0}^{g \rightarrow g} \right)^{-1} \mathbf{P}_{c,vv}^g$$

- DRAGON evaluation is single precision.
- Matlab evaluation is double precision.



Verification

17

Errors:

- Complete CP matrix:

$$\epsilon_c = \left(\frac{||\mathbf{P}_{c,vv}^D - \mathbf{P}_{c,vv}^M||}{||\mathbf{P}_{c,vv}^M||} \right) < 1.0 \times 10^{-6}$$

- Scattering modified CP matrix:

$$\epsilon_W = \left(\frac{||\mathbf{W}_{vv}^D - \mathbf{W}_{vv}^M||}{||\mathbf{W}_{vv}^M||} \right) < 3.0 \times 10^{-6}$$

Consistent with single precision nature of DRAGON.



Verification

18

The FLU: Module:

- To solve the collision probability equations for the flux (case with external source).
- To evaluate the problem eigenvalue (case without external source).
- To simulate leakage in an infinite cell and to evaluate the diffusion coefficients.



Verification

19

Scattering modified CP solution:

- Sources:

$$Q_{m,i,k}^g = F_i^g + \sum_{h=1, h \neq g}^G \Sigma_{s,i}^{h \rightarrow g} \tilde{\phi}_{i,k-1}^h$$

- Direct flux solution:

$$\phi_{i,k}^g = \sum_{j=1}^N W_{ij}^g Q_{j,k}^g$$

- Flux after rebalancing α^g :

$$\tilde{\phi}_{i,k}^g = \alpha^g \phi_{i,k}^g$$



Verification

20

DRAGON/Matlab comparison for T3D:

Iteration k	Group g	Matlab-DRAGON ($\times 10^7$)		
		Q	ϕ	$\tilde{\phi}$
1	1	1.0636	2.4653	19.0556
1	2	1.6990	2.1748	2.8039
2	1	1.0934	2.9636	18.1880
2	2	1.3440	2.1592	5.7258

Consistent with single precision nature of DRAGON.



Verification

21

The EDI: Module:

- To perform region homogenization of microscopic and macroscopic cross sections.
- To perform group condensation of microscopic and macroscopic cross sections.
- To compare the results from two different executions including:
 - The evaluation of the errors in reaction rates and output cross sections from two different executions.
 - The evaluation of incremental cross sections.



Homogenization and condensation procedure:

● Flux:

$$\phi_I^H = \frac{1}{V_I} \sum_{i \in I} \sum_{h \in H} \phi_i^h V_i$$

● Total cross section:

$$\Sigma_I^H = \frac{1}{V_I \phi_I^H} \sum_{i \in I} \sum_{h \in H} \phi_i^h V_i \Sigma_i^h$$

● Scattering cross section:

$$\Sigma_{I,s}^{H \rightarrow G} = \frac{1}{V_I \phi_I^H} \sum_{i \in I} \sum_{h \in H} \sum_{g \in G} \phi_i^h V_i \Sigma_{i,x}^{h \rightarrow g}$$



Verification

23

DRAGON/Matlab comparison for T3D:

$(\times 10^7)$	Case 1	Case 2	Case 3
$\delta \Sigma^1$	0.0401	-0.5698	-0.1446
$\delta \Sigma^2$	-0.2713		
$\delta \Sigma_s^{2 \rightarrow 1}$	-0.2232	-0.4730	-0.0075
$\delta \Sigma_s^{1 \rightarrow 1}$	0.0311		
$\delta \Sigma_s^{2 \rightarrow 2}$	-0.6877		
$\delta \Sigma_s^{1 \rightarrow 2}$	0.1251		

- Case 1 is for full homogenization/no condensation.
- Case 2 is for no homogenization/full condensation.
- Case 3 is for full homogenization/full condensation.



Verification

24

Conclusions of the verification exercise:

- No major error detected in DRAGON using the verification process.
- Output file does not always contain adequate information. This lead to the revision of some output routine to produce more significant information.
- Contents of the data structures not always correctly documented. This lead to revisions in the DRAGON documentation.



Verification

25

- Most of the major bugs in DRAGON were resolved using DRAGON self-verification procedures.
- Some bugs were resolved using DRAGON over-programmation (in EXCELT: and EXCELL: modules for example).
- Most of the errors remaining in DRAGON are detected by users.



Validation

1

Early DRAGON validation against experimental results:

- Experimental database provided by Chalk River Laboratory.
- Limited to 2-D zero-burnup transport calculations.
- Contains experimental information for a large number of important lattice parameters.
- Compiled for a variety of fuel and moderator types.



Validation

2

Evaluation of the performance of DRAGON by comparison with benchmark results:

1. NEACRP benchmark BWR lattice cell problems.
2. MOX Fueled VVER Cell Benchmarks.
3. 2-D and 3-D VENUS-2 MOX benchmark.



Validation

3

DRAGON validation against experimental results:

- Uses WIMS–AECL format Winfrith microscopic cross section library.
- Input data provided in the form of WIMS–AECL input data files.
- Only most reliable or default options of DRAGON considered.
- Cluster geometries in an hexagonal or square cell.
- Radial and axial buckling provided.



Validation

4

DRAGON Calculation Options:

- The explicit geometry used for self shielding.
- A discretized geometry used for solving the transport equation.
- Moderator, coolant and fuel regions divided into sub-regions of equal thickness not exceeding 0.4 cm.
- Test material are also self-shielded if required.
- Microscopic reaction rates for each test isotope in each region generated once flux distribution is known.



Validation

5

- Isotropic collision probability technique (standard P_{ij} method) in full 2-D geometry.
- Geometry analyzed for each test case and best compromise between precision and calculation time for tracking selected.
- Homogeneous B_1 leakage model with total buckling considered $B^2 = B_r^2 + B_z^2$.
- Heterogeneous B_1 leakage model with radial and axial buckling also considered.



Validation

6

Validation Parameters:

- Spatial dependence of flux (region i):

$$\rho_i = V_{\text{fuel}} \frac{\sum_g \sigma_{a,I}^g \phi_i^g}{\sum_{k \in \text{fuel}} V_k \sum_g \sigma_{a,I}^g \phi_k^g}$$

- Select isotope I such that $\sigma_{a,I}^g$ behaves as $1/v$ (^{55}Mn or ^{63}Cu for example).



Validation

7

- Neutron energy spectrum (region i):

$$N_i^{I/J} = \frac{\sum_g \sigma_{a/f,I}^g \phi_i^g}{\sum_g \sigma_{a/f,J}^g \phi_i^g}$$

- Select isotope I and J such that:

$$\sigma_{a/f,I}^g / \sigma_{a/f,J}^g = \alpha(1 + \beta \delta_{gh})$$

to enhance artificially group h . Consider:

- Ratio of ^{239}Pu to ^{235}U fission.
- Ratio of ^{176}Lu to ^{55}Mn absorption.
- Ratio of ^{115}In to ^{55}Mn absorption.



Validation

8

- Conversion ratio for cell with Uranium fuel:

$$C_i^{\text{fuel}} = \frac{\int v \sigma_a^{238\text{U}}(v) \phi_i(v) dv}{\int v \sigma_f^{235\text{U}}(v) \phi_i(v) dv}$$

- Reactivity of cell.
- Void effect.
- Fast fission ratio.



Validation

9

- Up to 52 parameters can be compared.
- All the parameters are not necessarily available for all experiments.
- The results are presented in the form of absolute errors or absolute error.



Validation

10

Results for k_{eff} :

- B_1^{hom} : systematic error of 1.48 ± 3.63 .
- B_1^{het} : systematic error of -1.0 ± 2.83 .



Validation

11

Relative errors on k_{eff} :

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	All	197	3.64 ± 7.26	1.15 ± 5.16
	D ₂ O	61	4.40 ± 7.73	1.04 ± 5.42
	Void	75	3.63 ± 7.82	0.88 ± 5.56
	Organic	39	3.97 ± 6.96	2.34 ± 4.99
	H ₂ O	20	0.90 ± 2.94	0.26 ± 2.75



Validation

12

Relative errors on k_{eff} :

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
UO ₂	All	96	7.21±8.80	3.43±6.08
	D ₂ O	35	7.86±8.66	3.24±6.05
	Void	39	7.34±9.14	3.39±6.23
	Organic	17	7.50±9.32	4.83±6.56
	H ₂ O	5	0.63±2.32	0.37±2.35



Validation

13

Relative errors on k_{eff} :

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
PuUO ₂	All	57	-0.35±2.36	-1.39±2.50
	D ₂ O	13	-0.08±1.38	-1.47±1.94
	Void	23	-0.65±2.95	-1.80±3.18
	Organic	10	-0.06±1.48	-0.61±1.60
	H ₂ O	9	-0.69±2.86	-1.39±2.36



Validation

14

Results for void effect:

- B_1^{hom} : systematic error of -0.68 ± 1.05 .
- B_1^{het} : systematic error of -0.5 ± 0.89 .
- B_1^{het} slightly better than B_1^{hom} .



Validation

15

Relative errors on void effect:

Fuel	N	B_1^{hom}	B_1^{het}
		$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	55	-0.68 ± 1.05	-0.50 ± 0.89
UO ₂ (N)	28	-1.31 ± 1.10	-0.95 ± 0.99
UO ₂ (D)	3	0.18 ± 0.49	0.18 ± 0.49
U-Metal	6	-0.06 ± 0.60	-0.04 ± 0.60
UC	5	-0.56 ± 0.23	-0.49 ± 0.21
U ₃ Si	2	-0.32 ± 0.31	-0.26 ± 0.27
PuUO ₂	11	0.20 ± 0.17	0.16 ± 0.24



Validation

16

Results for neutron spatial density ratio:

- Results nearly independent of leakage model.
- For most cases $|E^r| < 1$ in fuel.
- Error is more important at cell edge.



Validation

17

Relative errors for neutron density inner fuel pin:

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	All	75	0.25 ± 0.65	0.27 ± 0.63
	D ₂ O	20	0.50 ± 0.55	0.50 ± 0.55
	Void	26	0.38 ± 0.49	0.37 ± 0.49
	Organic	20	0.00 ± 0.82	0.07 ± 0.78
	H ₂ O	9	-0.11 ± 0.62	-0.05 ± 0.59



Validation

18

Relative errors for neutron density cell edge:

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	All	77	0.26 ± 1.37	0.25 ± 1.34
	D ₂ O	20	0.32 ± 0.85	0.38 ± 0.85
	Void	28	0.19 ± 1.12	0.35 ± 1.10
	Organic	20	1.00 ± 1.41	0.78 ± 1.33
	H ₂ O	9	-1.27 ± 1.77	-1.52 ± 1.66



Validation

19

Results for neutron energy distribution:

- Results nearly independent of leakage model.
- $|E^r| < 1$ in fuel.
- $|E^r| < 1$ at cell edge except for H₂O cooled cells.



Validation

20

Relative errors for fuel average $^{239}\text{Pu}/^{235}\text{U}$ fissions:

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	All	4	-0.12 ± 1.34	-0.10 ± 1.34
	D ₂ O	10	0.29 ± 1.69	0.30 ± 1.69
	Void	14	-0.07 ± 1.20	-0.08 ± 1.20
	Organic	7	0.03 ± 1.66	0.09 ± 1.66
	H ₂ O	11	-0.62 ± 0.88	-0.59 ± 0.90



Validation

21

Relative errors for cell edge $^{239}\text{Pu}/^{235}\text{U}$ fissions:

Fuel	Coolant	N	B_1^{hom}	B_1^{het}
			$E^r \pm \sigma^r$	$E^r \pm \sigma^r$
All	All	6	-0.75 ± 1.32	-0.74 ± 1.32
	D ₂ O	2	-0.09 ± 0.05	-0.10 ± 0.05
	Void	2	-0.91 ± 1.07	-0.91 ± 1.07
	Organic	0		
	H ₂ O	2	-1.24 ± 2.49	-1.21 ± 2.51



Validation

22

Conclusions of early validation attempt:

- Results are relatively good for most cases.
- Major problem are for relative conversion ratio in individual fuel pins (Probablyly due to ^{238}U self-shielding).
- Gain from B_1^{hom} to B_1^{het} not considerable except possibly for k_{eff} and void effect.
- Other libraries still remain to be tested.



Validation

23

MOX Fueled VVER Cell Benchmarks.

- Hexagonal cell.
- Various mixture composition including 2 types of MOX fuel.
- Local parameters variations.
- Burnup dependent.
- Results available:
 - ENDF/B-V based SAS2H calculations.
 - ENDF/B-VI based HELIOS calculations.



Validation

24

Geometry:

- Hexagonal cell with $l = 1.275$ cm lattice pitch.
- Central fuel region with Zirconium fuel clad and light water moderator.

Neutron leakage:

- k_{∞} for cell without leakage.
- k_{eff} for cell with imposed leakage (0.003 cm^{-2}).
- k_0 for critical cell with buckling search.



Validation

25

8 different fuel compositions including:

- Variant V1: 4.4% enriched LEU fuel.
- Variant V2: MOX from weapon grade plutonium.
- Variant V3: Spent 4.4% LEU fuel with fission products.
- Variant V10: MOX from reactor grade plutonium.



Validation

26

Local parameters for various operating conditions:

- State S1 for normal operating conditions: Fuel at 1027 K; Cladding+moderator at 579 K; ^{135}Xe and ^{149}Sm at saturation; 600 ppm of boron in moderator.
- State S3: S1 without boron.
- State S4: S1 without ^{135}Xe and ^{149}Sm .
- State S5: S4 with fuel+cladding+moderator at 579 K.
- State S6: S4 with fuel+cladding+moderator at 300 K.



Validation

27

Depletion calculations:

- All local parameters in state S1.
- Fuel variants V1 with $P = 41.9109$ kW/kg.
- Fuel variants V2 with $P = 41.7837$ kW/kg.
- Fuel variants V10 with $P = 34.0078$ kW/kg.
- Maximum burnup step of 2 MWd/kg.
- Final burnup of 60 MWd/kg.



Validation

28

Libraries for DRAGON:

- ENDF/B-V and ENDF/B-VI based 89 group WIMS-AECL library.
- No curium isotopes in ENDF/B-V library.
- No self-shielding for ^{241}Pu , ^{242}Pu and ^{241}Am in ENDF/B-VI library.

2-D hexagonal pin-cell calculation for DRAGON using CP method.



Validation

29

Overall comparison:

$$\Delta k^{\text{Code}}(mk) = 1000 \times \left(\frac{1}{k^{\text{HELIOS}}} - \frac{1}{k^{\text{Code}}} \right)$$

with:

- Code=DRGE5 represents DRAGON with ENDF/B-V WIMS-AECL library.
- Code=DRGE6 represents DRAGON with ENDF/B-VI WIMS-AECL library.



Validation

30

Δk_0 (mk):

Variant	Code	State				
		S1	S3	S4	S5	S6
V1	SAS2H	-10.3	-10.1	-9.6	-8.5	-4.9
	DRGE5	-9.7	-10.3	-9.4	-8.2	-4.9
	DRGE6	-2.5	-3.4	-2.5	-2.4	-2.6
V2	SAS2H	-9.4	-9.4	-9.0	-7.6	-3.7
	DRGE5	-6.2	-6.3	-5.8	-4.8	-1.9
	DRGE6	-0.4	-0.7	-0.3	-0.3	-1.0
V3	SAS2H	-11.5	-11.3	-10.8	-9.7	-5.1
	DRGE5	-4.3	-4.5	-3.5	-3.2	-0.9
	DRGE6	-3.9	-4.1	-3.3	-4.1	-4.2
V10	SAS2H	-16.1	-16.4	-16.0	-15.6	-10.9
	DRGE5	-22.4	-22.3	-22.1	-23.1	-17.1
	DRGE6	-8.9	-9.0	-8.7	-10.9	-10.8



Validation

31

Comments:

- SAS2H average error -12.7 mk.
- DRGE5 average error -10.3 mk.
- DRGE6 average error -3.7 mk.
- Main differences between HELIOS and DRGE6:
 - due to the absence of resonance tables for some actinides (fuel variants V3 and V10).
- $\Delta k_{eff} - \Delta k_0 \approx 4.0$ mk for V2 which indicates that ^{239}Pu has a large effect on leakage.



Reactivity effects comparison:

$$\delta \rho_0^{\text{Code}}(\mathbf{S}_i - \mathbf{S}_j) = 100 \times \left(\frac{\rho_0^{\text{Code}}(\mathbf{S}_i - \mathbf{S}_j) - \rho_0^{\text{HELIOS}}(\mathbf{S}_i - \mathbf{S}_j)}{\rho_0^{\text{HELIOS}}(\mathbf{S}_i - \mathbf{S}_j)} \right)$$

$$\rho_0(\mathbf{S}_i - \mathbf{S}_j) = 1000 \times \left(\frac{1}{k_0(\mathbf{S}_i)} - \frac{1}{k_0(\mathbf{S}_j)} \right)$$

- S3-S1 → reactivity worth of 600 ppm of boron.
- S4-S1 → reactivity worth of saturating fission products.
- S5-S4 → the Doppler effect in the fuel.
- S6-S5 → uniform temperature increase in the cell including the effect of coolant density and boron concentration reduction.



Validation

33

$\delta\rho_0$ (%):

Variant	Code	State			
		S3-S1	S4-S1	S5-S4	S6-S5
V1	SAS2H	0.6	2.4	11.6	17.3
	DRGE5	-1.8	1.2	12.2	15.8
	DRGE6	-2.5	-0.1	1.0	-0.9
V2	SAS2H	0.2	3.3	10.8	8.9
	DRGE5	-0.2	3.6	7.6	6.5
	DRGE6	-1.8	0.9	0.2	-1.7
V3	SAS2H	0.5	2.2	7.3	13.8
	DRGE5	-0.5	2.5	2.5	6.8
	DRGE6	-0.7	1.7	-5.6	-0.4
V10	SAS2H	-2.6	1.8	3.1	9.2
	DRGE5	1.1	6.2	-6.5	12.0
	DRGE6	-1.2	2.6	-14.6	0.1



Validation

34

Comments:

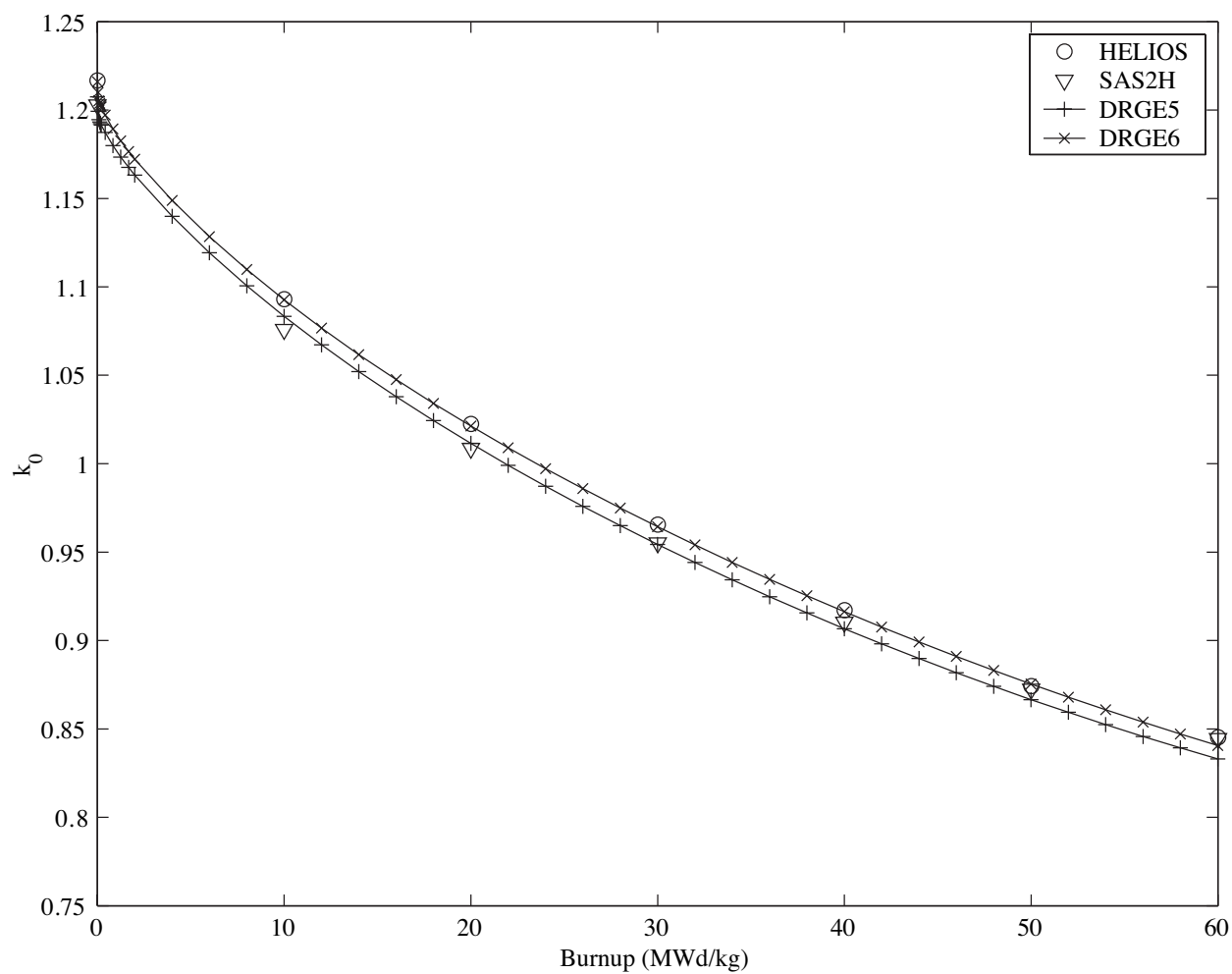
- Reactivity worth of boron predicted to within 4.7 %.
- Reactivity worth of Xe and Sm predicted to within 6.2 %.
- Reactivity worth of fuel Doppler effect predicted to within 18 %.
- Reactivity worth of uniform fuel temperature change predicted to within 37 %.



Validation

35

Depletion for fuel variant V2:

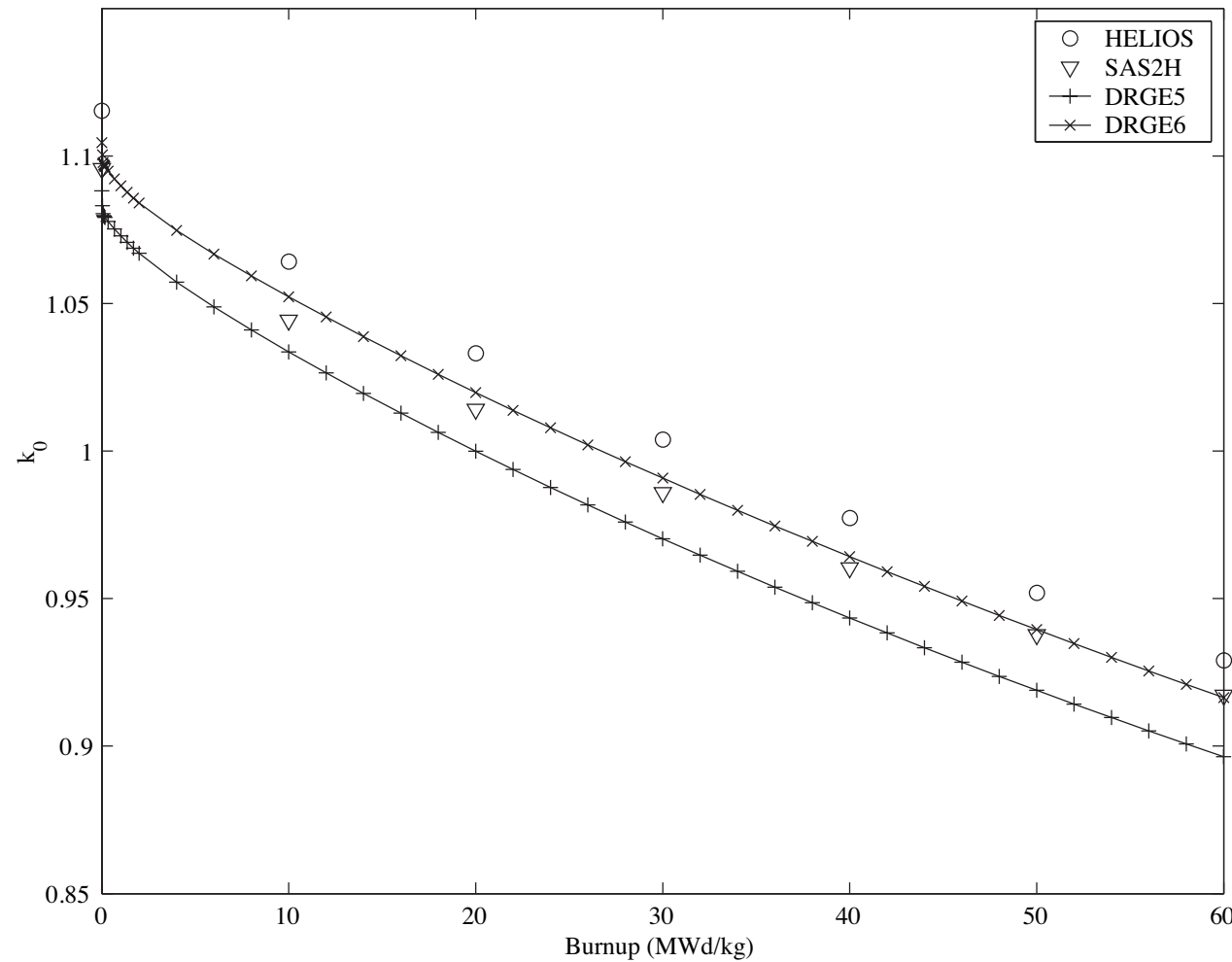




Validation

36

Depletion for fuel variant V10:





Validation

37

^{235}U concentration for fuel V2 (10^{-5} barns $^{-1}$ cm $^{-1}$):

B	HELIOS	SAS2H	DRGE5	DRGE6
0.0	3.84	3.84	3.84	3.84
10.0	3.17	3.17	3.17	3.18
20.0	2.56	2.55	2.56	2.57
30.0	2.01	2.01	2.01	2.02
40.0	1.53	1.55	1.54	1.55
50.0	1.14	1.17	1.15	1.15
60.0	0.824	0.869	0.83	0.835



Validation

38

^{239}Pu concentration for fuel V2 (10^{-4} barns $^{-1}$ cm $^{-1}$):

B	HELIOS	SAS2H	DRGE5	DRGE6
0.0	6.59	6.59	6.59	6.59
10.0	5.12	5.16	5.16	5.15
20.0	3.97	4.08	4.06	4.03
30.0	3.13	3.30	3.24	3.20
40.0	2.55	2.76	2.68	2.62
50.0	2.17	2.42	2.39	2.23
60.0	1.93	2.20	2.05	1.97



Validation

39

^{135}Xe concentration for fuel V2 (10^{-8} barns $^{-1}$ cm $^{-1}$):

B	HELIOS	SAS2H	DRGE5	DRGE6
0.0	0.94	0.946	0.946	0.946
10.0	1.57	1.55	1.53	1.39
20.0	1.42	1.42	1.39	1.26
30.0	1.29	1.31	1.27	1.15
40.0	1.18	1.22	1.16	1.05
50.0	1.08	1.14	1.07	0.962
60.0	1.01	1.02	0.996	0.889



Validation

40

^{149}Sm concentration for fuel V2 (10^{-7} barns $^{-1}$ cm $^{-1}$):

B	HELIOS	SAS2H	DRGE5	DRGE6
0.0	0.737	0.737	0.737	0.737
10.0	1.79	1.74	1.76	1.88
20.0	1.75	1.74	1.74	1.88
30.0	1.66	1.68	1.66	1.77
40.0	1.56	1.60	1.56	1.63
50.0	1.46	1.52	1.46	1.49
60.0	1.37	1.31	1.36	1.37



Validation

41

Conclusions:

- ENDF/B-VI based DRAGON results are consistent with HELIOS.
- Main problem is for fuel variants that contain isotopes without resonance tables in the library.
- For depletion calculations, there seems to be a problem with the ^{135}Xe and ^{149}Sm concentration at high burnup.



Documentation

1

General documents for DRAGON 3.04B:

- User's guide (IGE-174).
- A guide describing the contents of the DRAGON data structures (IGE-232).
- Geometry numbering in DRAGON (IGE-233).
- The first part of the DRAGON theory manual (IGE-236).
- Programmer's guide (IGE-251).



Documentation

2

GANLIB and CLE-2000 reports:

- The GAN generalized driver (IGE-158).
- The CLE-2000 Toolbox (IGE-163).



Documentation

3

Restricted DRAGON report:

- DRAGON validation report (École Polytechnique/AECL report).
- DRAGON verification report for AECL (École Polytechnique/AECL report).



Documentation

4

Other documentations:

- A few reports comparing DRAGON with benchmarks.
- A large number of articles describing:
 - The methods implemented in DRAGON.
 - Comparison of DRAGON with benchmarks.
 - Comparison of DRAGON with other codes.
 - Use of DRAGON for modeling different problems.



Using DRAGON

1

Contents:

- The DRAGON input file.
- The DRAGON modules.
- The DRAGON data structures.
- Working with CLE-2000.
- Flow chart in DRAGON input decks.



Using DRAGON

2

Input file format:

- 72 columns.
- free format instruction.
- comments can be inserted in the input deck using:
 - * in first column.
 - ! anywhere.
- end of input for a command or instruction is indicated by:
 - ;



Using DRAGON

3

A typical input file for DRAGON contains instructions to:

- Define the modules that will be used.
- Define the data structures that are required and specify their format.
- Define local variables required to control the execution.
- Define the local procedures required for execution.
- Import the files required for the execution.
- Describe the sequence of modules to execute with their input data.
- Clean up, export files and finish the execution.



Modules

1

The DRAGON cross section generation modules:

- **MAC :** module used to generate or modify a DRAGON MACROLIB.
- **LIB :** module used to generate or modify a DRAGON MICROLIB. It can read different types of microscopic cross-section libraries. Cross sections are interpolated in temperature and dilution. It can access microscopic cross-section libraries in the formats DRAGLIB, MATXS, WIMS-D4 and WIMS-AECL. This module also produces a MACROLIB that is compatible with that generated with the **MAC :** module.

The spatial location of the mixtures defined in **MAC :** and **LIB :** will be defined using the **GEO :** module.



Modules

2

The DRAGON geometry analysis modules:

- **GEO**: module used to generate or modify a geometry.
- **JPMT**: the tracking module for approximate J_{\pm} CP calculations.
- **SYBILT**: the tracking module for approximate Interface Current CP calculations.
- **EXCELT**: the tracking module for the standard 2–D and 3–D CP calculations and for 2–D MOC calculations. This is the most precise and general tracking procedure in DRAGON.
- **BIVACT**: a 2–D diffusion tracking module used for homogenization purpose.



Modules

3

The CP flux calculation modules:

- **ASM:** module which uses the tracking information to generate a multigroup response or collision probability matrix.
- **FLU:** module which uses the multigroup response or collision probability matrix to solve the transport equation for the flux.



Modules

4

The alternate CP evaluation module:

- EXCELL: module combines the tracking module EXCELT: and the assembly module ASM: thereby avoiding the generation of a binary tracking file for 3-D CP calculations.

The MOCC flux calculation module:

- MOCC: module to solve the transport equation for the flux using the cyclic method of characteristics.



Modules

5

Additional MICROLIB processing modules:

- SHI : used to perform resonance self-shielding calculations.
- EVO : used for isotopic depletion calculations.
- INFO : a module to compute the density and isotopic contents of heavy or light water and the isotopic contents of UO_2 or ThUO_2 fuels.



Modules

6

The main reaction rates processing modules:

- **EDI** : editing module for group condensation and spatial homogenization of cross sections based on reaction rates conservation. The direct flux/volume homogenization technique is programmed as well as the SPH method for homogenization by equivalence.
- **CPO** : module for building the standard burnup dependant reactor database used in DONJON.
- **CFC** : module for building the local parameter burnup dependent reactor database based on feedback coefficients that can be used in DONJON.



Modules

7

Some utility modules:

- MRG: module to pre-homogenize a geometry tracked using the module EXCELT:.
- PSP: module to generate PostScript images for 2-D geometries which can be tracked using the module EXCELT:.



Data structures

1

The formats of the data structures permitted are:

- **LINKED_LIST**: data structure stored in memory.
- **XSM_FILE** data structure stored on direct-access file.
- **SEQ_BINARY** data structure stored on sequential binary file.
- **SEQ_BINARY** data structure stored on sequential ASCII file.

The name of a data structure is case sensitive.



Data structures

2

LINKED_LIST: data structure stored in memory:

- Generally used to transfer information between modules (exception: the binary tracking file).
- Should be avoided for ASMPIJ data structure (because one may rapidly run into memory limitations).
- Should not be used if one wants to keep the information in the data structure once the execution is completed (for restart for example)
- It is currently the only format permitted for the TRACKING data structure.



Data structures

3

XSM_FILE: data structure stored on direct-access file:

- Can be used to transfer information between modules (exception: the binary tracking file and the TRACKING data structure).
- Should be used for ASMPIJ data structure (unless the number of regions and group in the problem are small).
- Should be used if one wants to keep the information stores the data structure once the execution is completed (for restart for example).
- Can be used to create a backup of a LINKED_LIST on a direct-access file.
- The contents of this file is system dependent.



Data structures

4

SEQ_BINARY: sequential binary file:

- Mainly used to store the binary tracking file.
- Can be used to create a backup of a `LINKED_LIST` or a `XSM_FILE` file on a sequential binary file.
- The contents of this file is system dependent.

SEQ_ASCII: sequential binary file:

- Generally used to create a backup of a `LINKED_LIST` or a `XSM_FILE` file on a sequential ASCII file.
- The contents of this file is system independent.
- This file can be viewed online using a simple text editor.



Data structures

5

A MACROLIB contains group ordered macroscopic cross-sections library:

- It can be a stand-alone structure or it can be included into a larger structure, such as a MICROLIB or an EDITION structure.
- It must be stored either in a linked list or an XSM file.
- It can be created by the MAC:, LIB: and EDI: modules.
- It can also be modified by the MAC:, LIB:, SHI: and EVO: modules.



Data structures

6

A MICROLIB contains a microscopic and a macroscopic cross-section library:

- It always contains a MACROLIB substructure.
- It can be a stand-alone structure or included into a larger structure, such as an EDITION structure.
- It must be stored either in a linked list or an XSM file.
- It can be created by the LIB: and EDI: modules.
- It can also be modified by the LIB:, MAC:, SHI: and EVO: modules.



Data structures

7

A GEOMETRY a contains "geometry" description:

- It can be a stand-alone structure or included into a larger structure, such as another GEOMETRY structure.
- It must be stored either in a linked list or an XSM file.
- It can be created and modified by the GEO: module.



Data structures

8

A TRACKING contains the general geometry description, the result of the geometry analysis and the main tracking parameters:

- It is a stand-alone structure.
- It must be stored on a linked list.
- It can be created by the JPMT:, SYBILT:, EXCELT: and EXCELL:. modules.
- It can be modified by the MRG: modules.



Data structures

9

A ASMPIJ contains the multigroup response and collision probability matrices:

- It is a stand-alone structure.
- It must be stored on a linked list or an XSM file.
- It is created by the ASM: or EXCELL: modules.



Data structures

10

A FLUXUNK contains the multigroup fluxes and the problem eigenvalue:

- It is a stand-alone structure.
- It must be stored on a linked list or an XSM file.
- It is created by the `FLU:` and `MOCC:` modules.



Data structures

11

A EDITION contains condensed and/or merged microscopic and macroscopic cross-sections libraries:

- It is a stand-alone structure but can contain MACROLIB and MICROLIB substructure.
- It must be stored on a linked list or an XSM file.
- It is created by the EDI : module.



Data structures

12

A BURNUP contains time dependent isotopic concentration:

- It is a stand-alone structure.
- It must be store on a linked list or an XSM file.
- It is created by the `EVO:` module.



Data structures

13

A CPO contains a simple reactor database:

- It is a stand-alone structure.
- It must be store on a linked list or an XSM file.
- It is created by the CPO: module.



Data structures

14

A FBMXSDB contains a reactor database with feedback coefficients:

- It is a stand-alone structure.
- It must be store on a linked list or an XSM file.
- It is created by the CFC : module.



Working with CLE-2000

1

Local variable types:

- **INTEGER**
- **REAL**
- **DOUBLE**
- **STRING** 72 character long string variables enclosed between " "
- **LOGICAL** logical variable resulting from a comparison of defined using = **\$True_L** or **\$False_L**

The name of a variable is case sensitive.



Working with CLE-2000

2

Operations on variables:

- Arithmetic operations in RPN: **+**, **-**, *****, **/** and ******:

```
delta := b 2 ** 4 . a c * * -
```

- Unary operations: **COS**, **SQRT**, **ABS**, **NOT** and **LN**:

```
delta := delta SQRT
```

- Relational operations: **<**, **>**, **<>** and **<=**:

```
condition := a b <=
```

- Mixed mode operations: **R_TO_I**, **I_TO_R**, etc.:

```
real := integer I_TO_R ;
```



Working with CLE-2000

3

Assigning or evaluating variables:

REAL (variable names) **:=** (value) **;**

EVALUATE (variable names) **:=** (value) **;**

ECHO (variable names) **;**

Using variable as input/output data to a module:

● **<< a >>** the variable a is used as input data.

● **>> a <<** the variable a is used as output data.



Working with CLE-2000

4

IF / THEN / ELSE statement:

```
IF          (condition) THEN      (statements)
[ ELSEIF (condition) THEN      (statements)
  ELSE      (statements)
]
ENDIF      ;
```

WHILE and **REPEAT** statements:

```
WHILE (condition) DO      | REPEAT
      (modif condition)    |      (modif condition)
      (statements)         |      (statements)
ENDWHILE ;                | UNTIL (condition) ;
```



Working with CLE-2000

5

Input deck procedures:

- used to segment large input deck into sub-deck.
- the main input deck can be seen as a FORTRAN program while the sub-decks can be viewed as a subroutine.
- like in a subroutine the sub-decks can be programmed to use input variables.



Working with CLE-2000

6

Example of an input deck procedure (File PWRPinCell.c2m):

```
*-----
* Procedure Geometry in file PWRPinCell.c2m
* Voltrk TrackFile := PWRPinCell ::
*   <<LocPitch>> <<LocRadius>> <<LocSplitR>>
*   >>MaxReg<<   ;
* Procedure parameters
*-----
PARAMETER          Voltrk  TrackFile  ::
    ::: LINKED_LIST Voltrk          ;
    ::: SEQ_BINARY  TrackFile      ;   ;
REAL              LocPitch LocRadius          ;
INTEGER          LocSplitR          ;
:: >>LocPitch<< >>LocRadius<< >>LocSplitR<<   ;
```



Working with CLE-2000

7

```
*-----
*   Local information and procedure description
*-----
MODULE GEO: EXCELT: DELETE: END: ;
LINKED_LIST LocalGeo ;
INTEGER    MaxRegion  := Mesh 1 + ;
LocalGeo := GEO: :: CARCEL 1
    X- REFL 0.0 <<LocPitch>> X+ REFL
    Y- REFL 0.0 <<LocPitch>> Y+ REFL
    RADIUS 0.0 <<LocRadius>> MIX 1 2 SPLITR <<LocSplitR>> ;
Voltrk TrackFile := EXCELT: LocalGeo ::
    MAXR <<MaxRegion>>
    TRAK TISO 20 100.0 ;
LocalGeo := DELETE: LocalGeo ;
:: <<MaxRegion>> ;
END: ;
```




Working with CLE-2000

8

Calling the procedure PWRPinCell (File PWRPinCell.c2m):

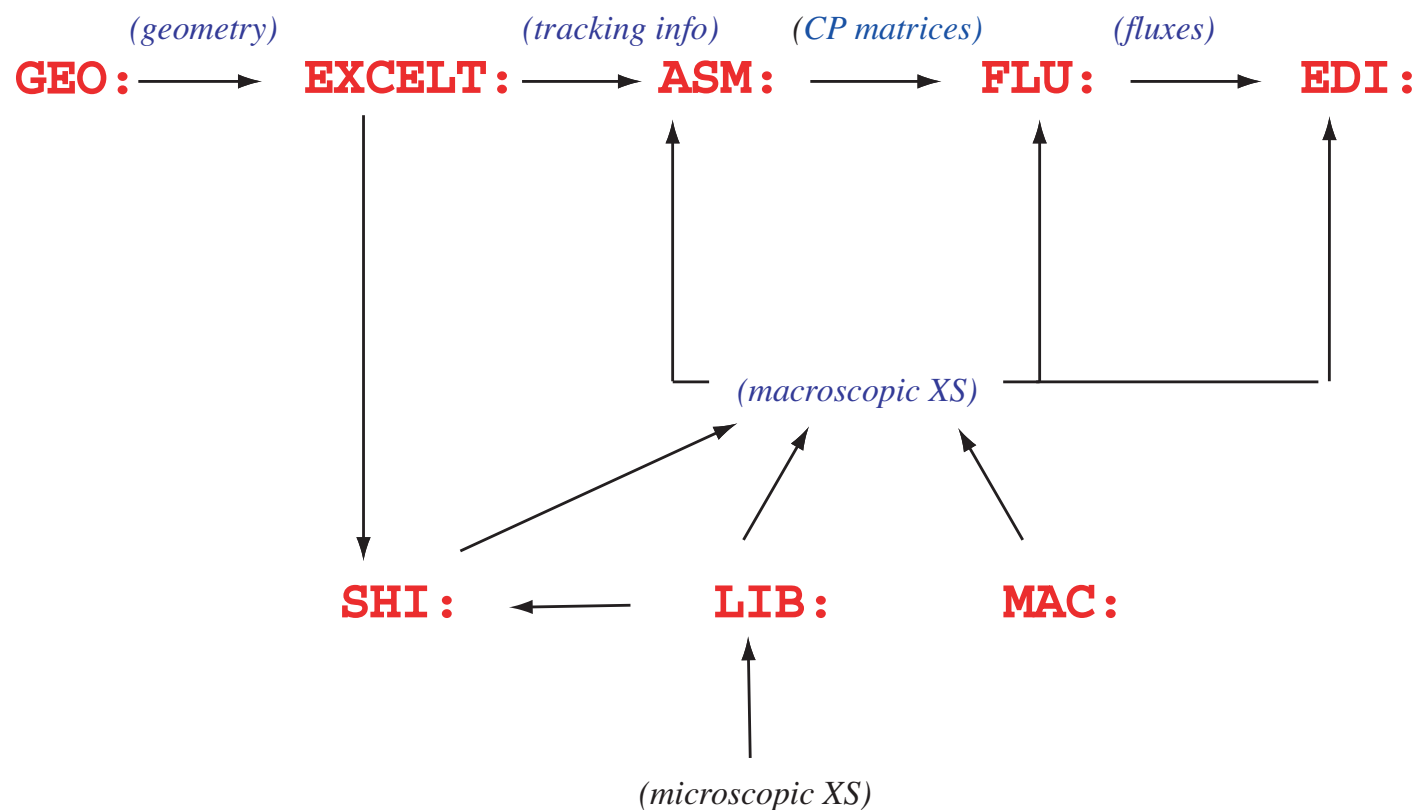
```
*-----
* Calling procedure PWRPinCell.c2m
* Voltrk TrackFile := PWRPinCell ::
*   <<LocPitch>> <<LocRadius>> <<LocSplitR>> >>MaxReg<<  ;
*-----
LINKED_LIST Voltrk          ;
SEQ_BINARY  TrackFile      ;
REAL        Pitch  Radius  := 1.2 0.4    ;
INTEGER     Mesh        := 5    ;
INTEGER     MaxReg       ;
PROCEDURE   PWRPinCell    ;
VolTrk Trackfile := PWRPinCell ::
    <<Pitch>> <<Radius>> <<Mesh>> >>MacReg<<  ;
ECHO "MaxReg =" MaxReg ;
VolTrk Trackfile := DELETE: VolTrk Trackfile ;
END: ;
```



DRAGON examples

1

Sequence of module for a straightforward DRAGON execution:

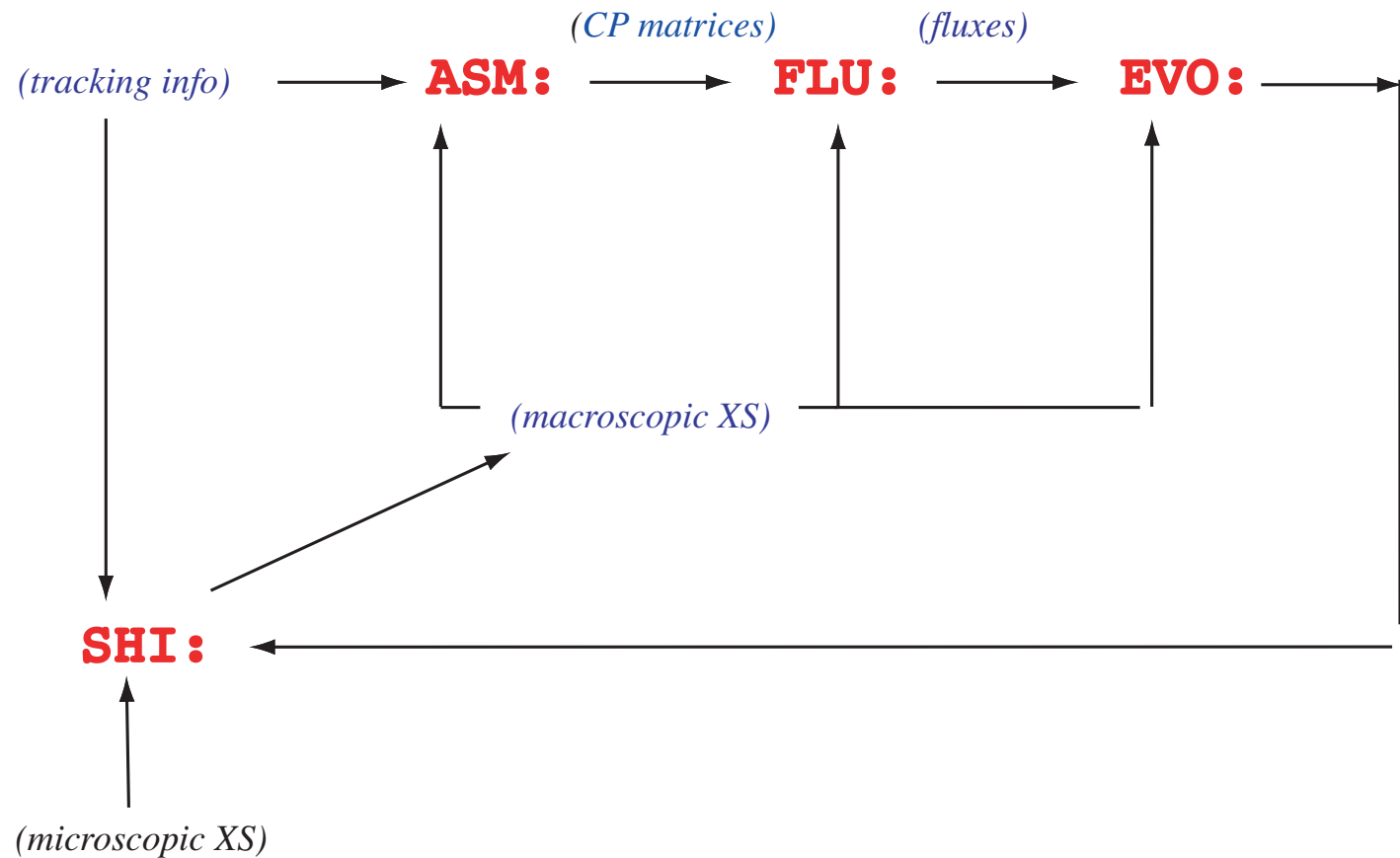




DRAGON examples

2

Sequence of module for a burnup cycle in DRAGON:

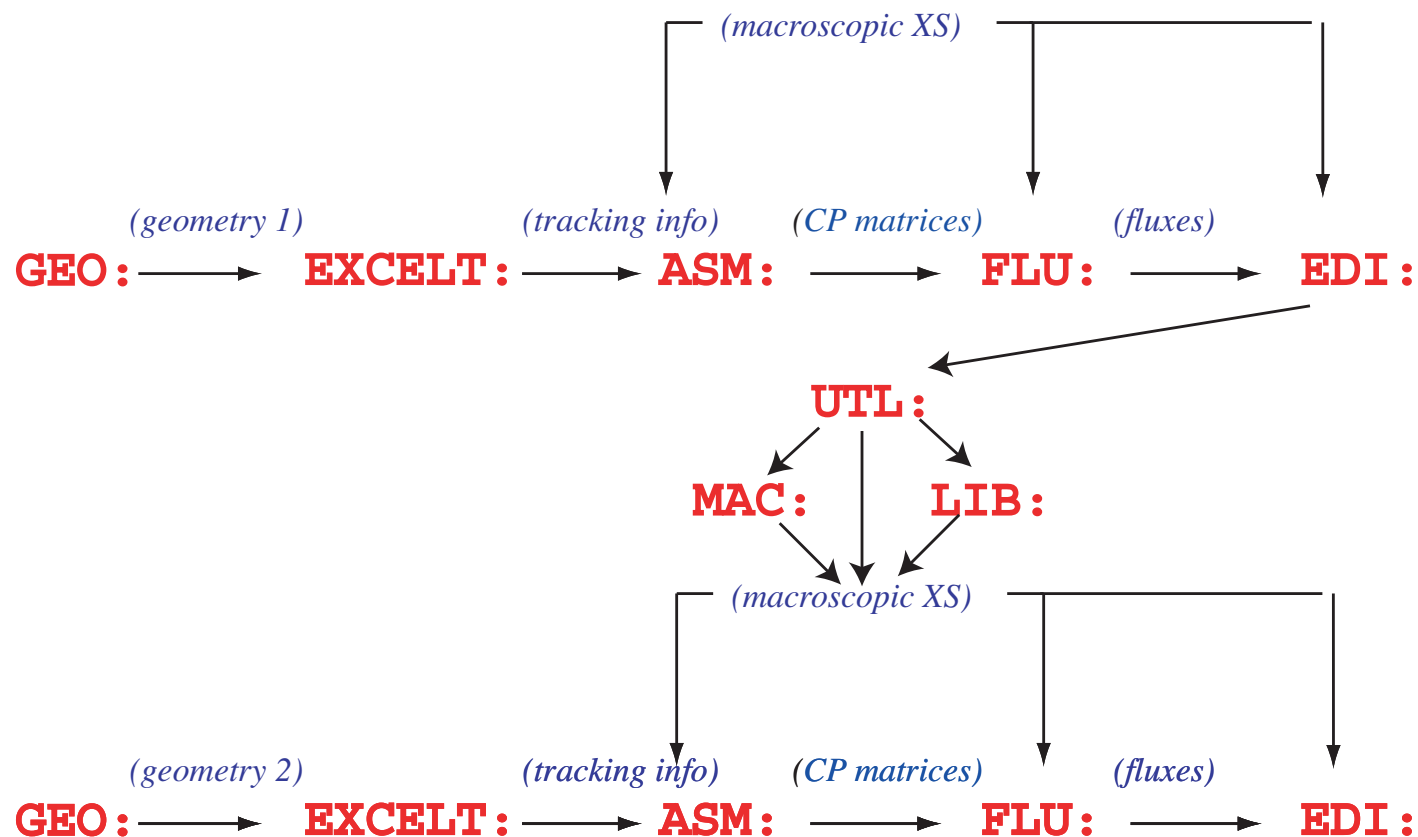




DRAGON examples

3

Multiple level DRAGON execution with homogenization:





DRAGON examples

4

Geometry validation and results analysis using the PSP module:

- **PSP**: generate PostScript representation of 2D geometry.
- Can use a GEOMETRY or TRACKING data structure for geometry description.
- Limited to simple 2-D valid EXCELT type geometry.
- Geometry can be colored by mixture number, region number or flux level (when FLUXUNK data structure is provided).



DRAGON examples

5

Tracking validation:

- Select quadrature densities and angular quadrature level.
- Verify volume and surface integration errors provided in output file (EDIT 2 in EXCELT: module).



DRAGON today and tomorrow

G. Marleau

Institut de génie nucléaire
École Polytechnique de Montréal



DRAGON today and tomorrow

1

Contents:

1. Advanced features.
2. Improvements in version 3.04 release R.
3. Limitations in DRAGON.
4. What to expect in version 3.05.
5. Towards DRAGON version 4.0.



Advance features

1

1. Cross-section perturbations.
2. Restart capabilities.
3. Double heterogeneity.
4. Geometry pre-homogenization.
5. Generating flux map in DRAGON.



Advance features

2

Cross-section perturbations using the `LIB:` module:

- MACROLIB data structure defined using the `LIB:` module can be perturbed in various ways:
 - Change the concentration of a single isotope in a mixture.
 - Change the density of a mixture.
 - Change the concentration of each fuel isotope using a BURNUP data structure.



Advance features

3

Example for burnup dependent fuel Doppler effect or CVR:

- Define a first MACROLIB with fuel at normal operating temperature (RefLib).
- Define a second MACROLIB with fuel at perturbed temperature (PerLib).
- Solve transport and depletion equation using RefLib and save burnup dependent isotopic concentrations ConcIso.
- Update PerLib at a specific time step using ConcIso.

```
PerLib := LIB: PerLib ConcIso :: BURN <<time>> ;
```



Advance features

4

Restart capabilities and MACROLIB combination using the MAC : module:

- The data structures generated by the various modules can all be saved (:=):
 - In a direct access XSM format for later use on the same system.
 - In a sequential ASCII format for later use on the same or a different system.
- The data structures can be exchanged between input files:
 - Create a MACROLIB during one execution.
 - Use this MACROLIB in various other execution.



Advance features

5

Example of CANDU incremental cross section evaluation:

- a) Create burnup dependent macroscopic cross section libraries:
 - Perform a time dependent CANDU 2-D cell calculation.
 - At selected time step (0, mid and full burnup), use `EDI` : to generate homogenized macroscopic cross sections and save.
 - Export the `EDITING` structure in `FuelEdi`.
 - Create using `LIB` : a `MACROLIB` that contain the structure materials and export in `STRCMAC`.



Advance features

6

- b) Create a MACROLIB for 3-D transport calculation:
 - Import in RDmac using MAC : the homogenized fuel properties at a specific burnup form FuelEdi.
 - Import in RDmac using MAC : the cross section associated with structure materials stored in STRCMAC.
 - Export RDmac.



Advance features

7

- c) Perform 3-D transport calculation:
 - Import RDmac.
 - Perform a 3-D CANDU supercell calculation.
 - Evaluate the incremental cross sections using the EDI : module.



Advance features

8

Example:

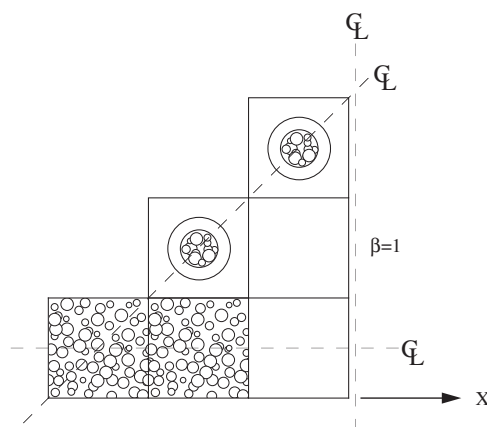
```
RDmac := UTL: RDmac ::  
    STEP UP 'FUEL1' ;  
MacroLib := MAC: RDmac ::  
    NMIX 17  
    MIX 1 1    MIX 2 2  
    MIX 3 3    MIX 4 4  
    ;  
RDmac := UTL: RDmac ::  
    STEP DOWN STEP UP 'Structure' ;  
MacroLib := MAC: MacroLib RDmac ::  
    NMIX 17  
    MIX 11 1  MIX 12 1  MIX 13 1  MIX 14 1  
    MIX 15 1  MIX 16 1  MIX 17 1  
    ;
```




Advance features

9

Geometry with double heterogeneity in DRAGON:



- The macro geometry is defined in terms of homogeneous mixtures and mixtures with structure (micro geometry).
- The micro geometry may be spherical (grains or pellets) or cylindrical (CANDU clusters).



Advance features

10

Double heterogeneity option in DRAGON:

1. Available only via the SYBILT: module.
2. All the processing is performed in the ASM: module via the sub-module SYBILP:.
3. Similar to the technique that A. Hébert implemented in the APOLLO-1 code as part of his Ph.D. work.



Advance features

11

Double heterogeneity treatment in DRAGON:

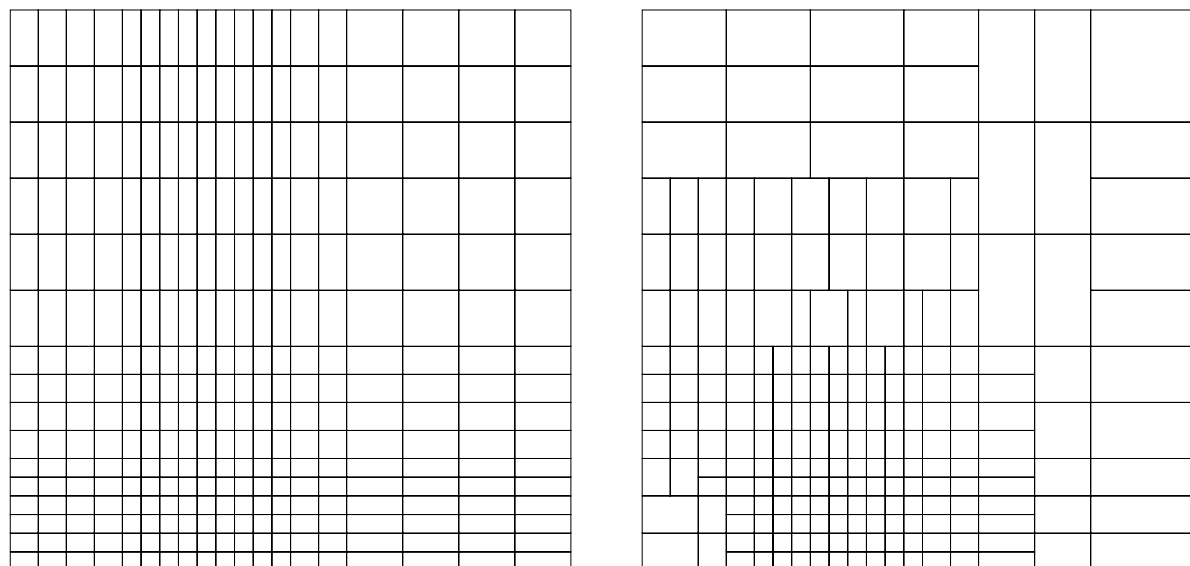
1. Compute the CP matrix for each micro geometry contributing to a DH mixture.
2. Use these CP matrix to generate an equivalent cross section for the DH mixtures.
3. Compute the CP matrix for the macro geometry.
4. Distribute the macro-geometry CP contribution in DH region on micro geometry.



Advance features

12

Non-uniform meshes in DRAGON:



- $N = 19 \times 16 \times 16 = 4864$ regions, $d_a = 100$ Mb/groups for uniform mesh.
- $N \approx 1500$ regions, $d_a = 9$ Mb/groups for non-uniform mesh.



Advance features

13

Generation of non uniform mesh in DRAGON using the MRG: module:

1. Define a geometry `RefGeo` with a uniform mesh.
2. Track this geometry with `EXCELT:` and generate the `VOLTRK` data structure `RefVol` and tracking files `RefTrk`.
3. Use the `MRG:` module to pre-homogenize `RefVol` and `RefTrk` into `MrgVol` and `MrgTrk`.
4. Use the `ASM:` module to compute the CP based on `MrgVol` and `MrgTrk`.



Advance features

14

Using the MRG: module:

```
MrgVol MrgTrk := MRG: MrgVol MrgTrk      ::  
  REGI    1    1    2    3    4    5    6    7    8  
          9   10   11   12   13   14   15   16   17   17  
          1    1    2   18   19   20   21   22   23  
        24   25   26   27   28   29   30   16   17   17  
        31   31   32   33   34   35   36   37   38  
        39   40   41   42   43   44   45   46   47   47  
        31   31   32   48   49   50   51   52   53  
        54   55   56   57   58   59   60   46   47   47  
        61   62   63   64   65   66   67   68   69  
        70   71   72   73   74   75   76   77   78   78  
  
    ...  
    ;  
PIJ := ASM: MacLib MrgVol MrgTrk ;
```

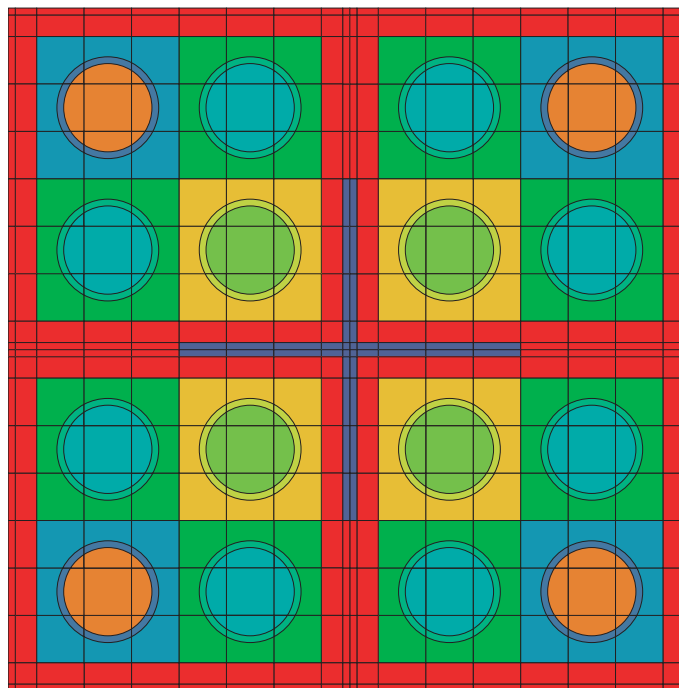


Advance features

15

A colored map by mixture number of a DRAGON BWR geometry:

Legend
Color by Mixture





Advance features

16

The `PSP` module works for most 2-D Geometry that can be processed by the `EXCELT` module to:

- Produce a colored map by mixture number in the geometry.
- Produce a colored map by region number in the geometry.
- Produce a colored map for the multigroup flux.

For validating 3-D geometry, proceed by projection of the 3-D geometry on various 2-D planes.

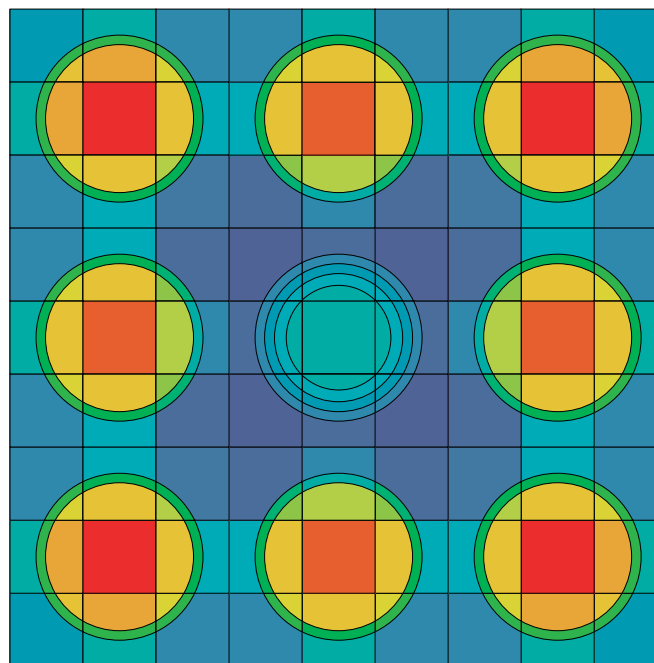


Advance features

17

A colored map for the multigroup flux in a BWR geometry:

Legend
Color by Flux in group 1
 $1.13E+01+(i-1)*5.78E-02 < Flux(i) \leq 1.13E+01+i*5.78E-02$





Advance features

18

The PSP : module can process:

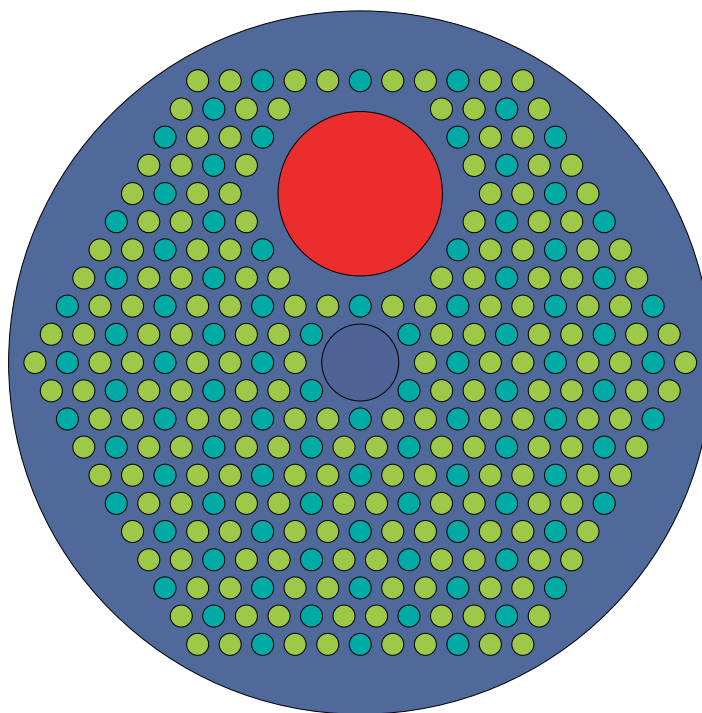
- An explicit DRAGON geometry. In this case the module PSP : calls the geometry analysis part of the EXCELT : module.
- A VOLTRK data structure produced by the EXCELT : module.
- A VOLTRK data structure produced by the MRG : module.



Advance features

19

A colored map by mixture number of a prismatic modular assembly (hexagonal cluster):





DRAGON version 3.04R

1

Updates in:

- GANLIB.
- UTILIB.
- DRAGON.

New modules:

- DRAGON.



DRAGON version 3.04R

2

Update to GABLIB/CLE-2000:

- String variable with blanks are now permitted as parameters of procedures.
- String variable length is now preserved.
- Mixed mode operations **I_TO_S** implemented.



DRAGON version 3.04R

3

Example of I_TO_S use:

```
STRING  Refname := "Test" ;
STRING  Dirname ;
INTEGER Step := 0 ;
WHILE Step 50 < DO
    EVALUATE  Step      := Step 1 + ;
    EVALUATE  Dirname := Refname Step I_TO_S + ;
    ...
    ...
    EditDS := EDI: EditDS Flux TmpMicLib TrackingF  ::
        COND 0.625 MERG COMP  SAVE ON <<Dirname>>      ;
ENDWHILE ;
```



DRAGON version 3.04R

4

Update to UTILIB:

- PSP drawing routines for `strike` and `fill` now combined with `gsave` to reduce dimensions of PostScript file.
- `XDRSET` and `XDRDBL` routine rationalized and generalized to other types of variable.
- `XDRCST` routine created to rationalized use of constants in DRAGON. (Avogadro number, neutron mass, etc.).



DRAGON version 3.04R

5

Update to DRAGON:

- The GEO: Module.
- The SYBILT: Module.
- The EXCELT: Module.
- The ASM: Module.
- The EXCELL: Module.
- The FLU: Module.
- The EDI: Module.
- The EVO: Module.



DRAGON version 3.04R

6

The GEO: module:

- Boundary condition SSYM is similar SYME but applied at the boundary of the cell or assembly.

The SYBILT: module:

- Large symmetric slab problems produced CP that were not symmetric.
- Both forward and backward tracking directions are now considered.



DRAGON version 3.04R

7

The EXCELT: module:

- The symmetry options can now be applied to isolated cells.
- The geometry analysis process is now controlled by a single routine and completely decoupled from the tracking process:
 - This was required to extend the capacity of the PSP:, EDI: modules.
 - It was also required to simplify the maintenance of the EXCELT: and EXCELL: modules.



DRAGON version 3.04R

8

For symmetric assemblies, the tracking is now automatically symmetrized:

- The symmetric tracking option was activated manually in 3.04B.
- Only some symmetries were considered.
- This lead to a reduction of a factor of 2 to 4 in size for some tracking file.

In addition:

- Tracks were being lost on Linux for 2-D and 3-D hexagonal assembly.
- Volume computation errors were found for some cluster geometries.



DRAGON version 3.04R

9

The ASM: module:

- Space requirements for the execution of this module is now provided.
- It is now possible to compute the CP matrix while reading the tracking file only once.

The EXCELL: module:

- The CP normalization technique did not work for explicitly voided regions.
- The renormalization technique were reprogrammed from the original routine in ASM: and the results cross checked.



DRAGON version 3.04R

10

The FLU : module:

- Adjoint and generalized adjoint flux algorithm is now available.
- For imposed leakage one can select imposed dB^2 or imposed B^2 with computed d .

The EDI : module:

- Homogenization per geometry option is now implemented.
- The MACROLIB and MICROLIB generated by the EDI : module are now complete and can be used for burnup calculation in DRAGON.



DRAGON version 3.04R

11

The EVO: module:

- The contents of the BURNUP data structure was incomplete.
- The isotopes names are now stored on the BURNUP data structure (required by the LIB: module for some executions).



DRAGON version 3.04R

12

New modules in DRAGON :

- The SAD: Module.
- The PER: Module.
- The HST: Module.



DRAGON version 3.04R

13

The SAD: module:

- Compute the generalized adjoint sources required for generalized perturbation theory methods.

The PER: module:

- Perform generalized perturbation theory calculations.



The HST : module:

- For history based diffusion/transport reactor calculations.
- It generates a HISTORY data structure:
 - It is a stand-alone structure.
 - It must be stored on a linked list or an XSM file.
 - It contains the local and global parameter associated with a series of fuel cells including their current isotopic contents.



Limitations in DRAGON

1

1. Geometry.
2. Memory.
3. Other.



Limitations in DRAGON

2

2-D Cartesian assembly in DRAGON:

- Cartesian mesh in each direction must extend to the whole geometry.
- A single cylinder per cell.
- Cylinders are by default centered in the cell but can be displaced using the OFFCENTER option.
- The Cartesian and annular mesh are superposed. Mixture must therefore be specified in R , then in x, y .
- Mixtures are specified even in location that do not exist.



Limitations in DRAGON

3

3-D Cartesian assembly in DRAGON:

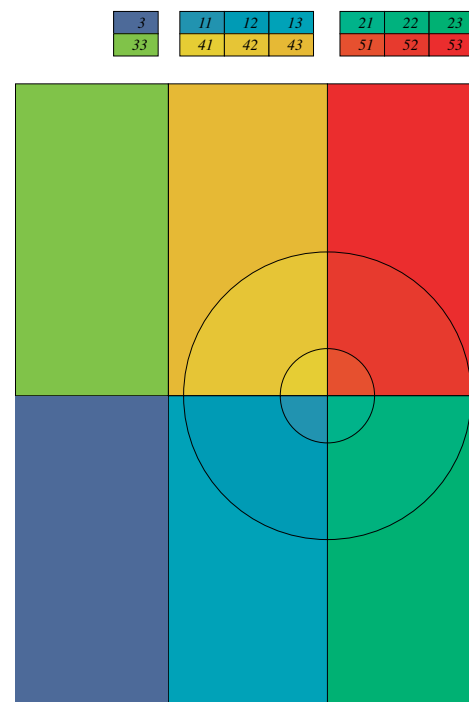
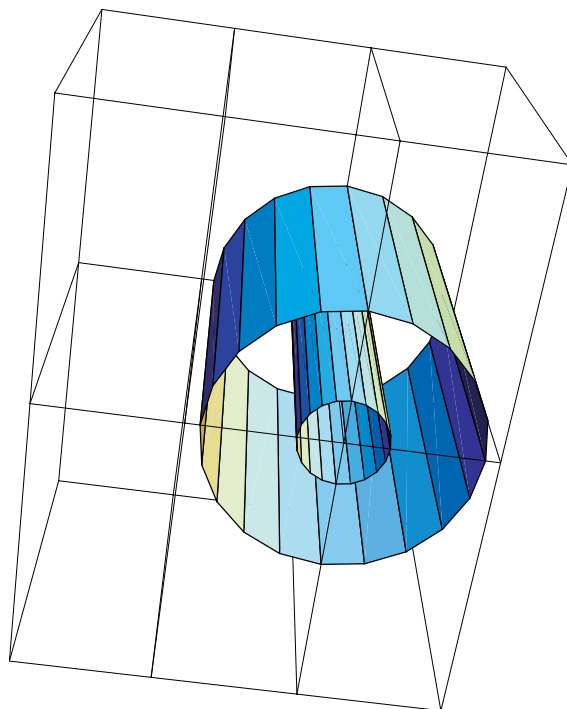
- Similar to 2-D assembly.
- Cylinders in different directions cannot intersect.
- Cylinders must extend to the whole geometry.
- Cylinders are by default centered in the cell. They can be displaced using the OFFCENTER option.



Limitations in DRAGON

4

A Simple 3-D Cell:





Limitations in DRAGON

5

Mixture specification for simple 3-D cell:

```
TMPGEO    := GEO:    :: CAR3D 1 1 1
CELL      FC1B
X- REFL X+ REFL Y- REFL Y+ REFL Z- REFL Z+ REFL
::: FC1B := GEO: CARCELZ 2 3 2 1
MESHX      <<MXLP>>    <<MXL>>    <<C14CN>>    <<MXYD>>
MESHY      <<MYLP>>    <<CALLCN>> <<PYLP>>
MESHZ      0.0          49.5
OFFCENTER  <<FC1XD>> <<FC1YD>>
RADIUS 0.0 <<RF2>>    <<RCT>>
MIX          1  2  3
           11 12 13
           21 22 23
           31 32 33
           41 42 43
           51 52 53 ; ;
```

Note: mixtures 1, 2, 31 and 32 not used.

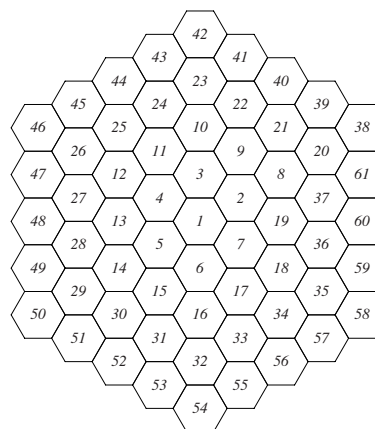


Limitations in DRAGON

6

2-D Hexagonal assembly in DRAGON:

- Either a single hexagonal cell or an assembly of hexagonal cell.
- Global boundaries of hexagonal assembly are jagged:





Limitations in DRAGON

7

2-D hexagonal assembly in DRAGON (continue):

- A single cylinder per cell.
- Cylinders are centered in the cell (no possibility to displace the center in this case).
- Sectors are possible (but not documented).

3-D hexagonal assembly in DRAGON:

- Similar to 2-D assembly.
- Cylinder completely fill the z direction.
- Sectors are possible (but not documented).

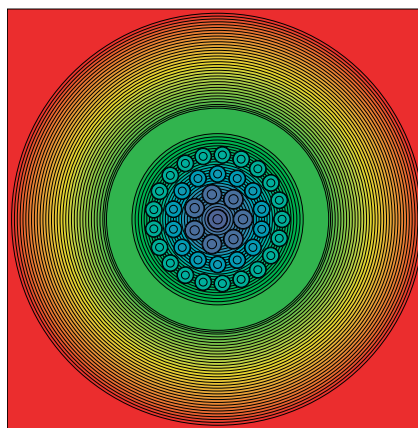


Limitations in DRAGON

8

Cluster in DRAGON:

- Not available in 3-D.
- A single cell permitted in 2-D.
- Pin clusters must be located inside the outer annular ring.
- Outer boundary can be Cartesian, hexagonal or annular.





Limitations in DRAGON

9

Exact boundary conditions:

- VOID: applied at the explicit boundary of the cell or assembly.
- SYME: applied at the center of the cells closest to the explicit assembly boundary specified.
- DIAG: applied at the center of the cells closest to the explicit assembly boundary specified.
- SSYM: applied at the explicit boundary of the cell or assembly.



Limitations in DRAGON

10

Approximate boundary conditions:

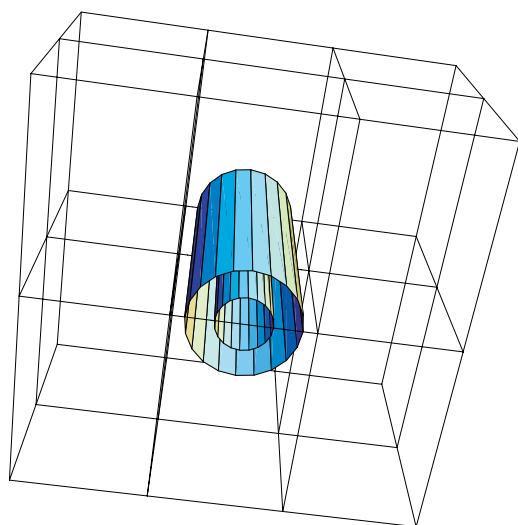
- REFL: applied at the explicit boundary of the cell or assembly. Mirror like reflections not available in 3-D.
- TRAN: applied at the explicit boundary of the cell or assembly. Mirror like translations not available in 3-D.
- ALBE: applied at the explicit boundary of the cell or assembly.



Limitations in DRAGON

11

Region identification for cells and assemblies:



Back

6	9	10
	<div style="border: 1px solid black; border-radius: 50%; padding: 2px; display: inline-block;"> 8 7 2 3 </div>	
1	4	5

Back

2	10	18
	<div style="border: 1px solid black; border-radius: 50%; padding: 2px; display: inline-block;"> 9 8 5 6 </div>	
1	7	17

Front

16	19	20
	<div style="border: 1px solid black; border-radius: 50%; padding: 2px; display: inline-block;"> 18 17 12 13 </div>	
11	14	15

Front

4	16	20
	<div style="border: 1px solid black; border-radius: 50%; padding: 2px; display: inline-block;"> 15 14 11 12 </div>	
3	14	19

one cell

three cells assembly



Limitations in DRAGON

12

Region identification for single cell:

- radially outward in a cell.
- from lower to upper x location in a cell.
- from lower to upper y location in a cell.
- from lower to upper z location in a cell.



Limitations in DRAGON

13

Region identification for one cell problem:

PLANE - 2

```
-----  
  ( 1)      16  ( 1)      19  ( 1)      20  
          ABSENT ( 1)      18          ABSENT  
          ABSENT ( 1)      17          ABSENT  
-----
```

```
  ( 1)      11  ( 1)      14  ( 1)      15  
          ABSENT ( 1)      13          ABSENT  
          ABSENT ( 1)      12          ABSENT  
-----
```

PLANE - 1

```
-----  
  ( 1)      6  ( 1)      9  ( 1)      10  
          ABSENT ( 1)      8          ABSENT  
          ABSENT ( 1)      7          ABSENT  
-----
```

```
  ( 1)      1  ( 1)      4  ( 1)      5  
          ABSENT ( 1)      3          ABSENT  
          ABSENT ( 1)      2          ABSENT  
-----
```



Limitations in DRAGON

14

Region identification for assembly of cells:

- Inside each cell as above.
- from lower to upper x cell location in the assembly.
- from lower to upper y cell location in the assembly.
- from lower to upper z cell location in the assembly.



Limitations in DRAGON

15

Region identification for three cells assembly:

PLANE - 2	-----					
	(1)	4	(1)	16	(1)	20
		ABSENT	(1)	15		ABSENT
		ABSENT	(1)	14		ABSENT

	(1)	3	(1)	13	(1)	19
PLANE - 1		ABSENT	(1)	12		ABSENT
		ABSENT	(1)	11		ABSENT

	(1)	2	(1)	10	(1)	18
		ABSENT	(1)	9		ABSENT
		ABSENT	(1)	8		ABSENT

	(1)	1	(1)	7	(1)	17
		ABSENT	(1)	6		ABSENT
		ABSENT	(1)	5		ABSENT

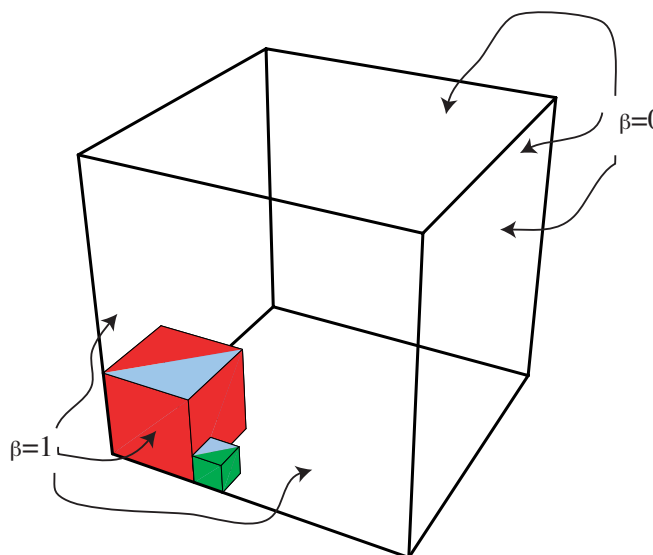


Limitations in DRAGON

16

Example of a simple 3-D problem.

- Total volume 1 liter ($V = 10^3 \text{ cm}^3$).
- Central fissile region is red ($V = 27 \text{ cm}^3$).
- Strong absorber is green ($V = 1 \text{ cm}^3$).

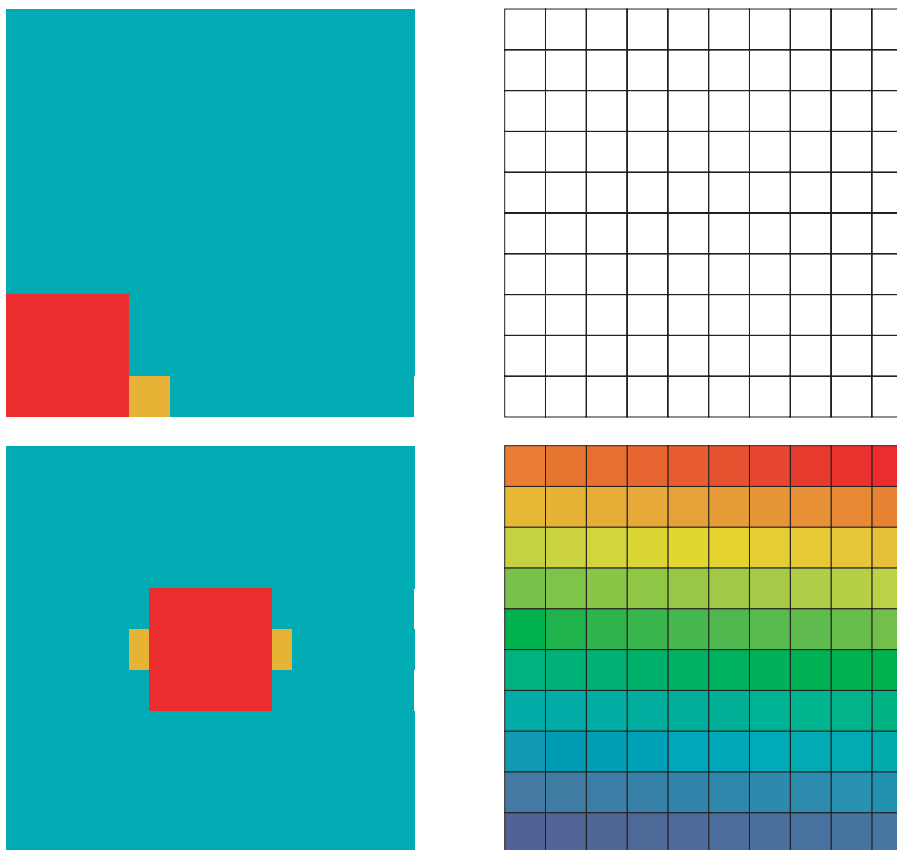




Limitations in DRAGON

17

Example of mesh for simple 3-D problem:



Note: $x - y$ and $x - z$ planes are identical.



Limitations in DRAGON

18

Select transport calculation mesh:

- Use symmetry whenever possible.
- For regions with strong absorber and fission:
 - Try to avoid using approximate boundary conditions.
 - Use fine mesh discretization.
- For moderator region:
 - Fine to coarse mesh discretization may be required.



Limitations in DRAGON

19

Select CP quadrature:

- As many angles as possible.
- Tracking density must be as dense as possible.
- Integration lines are renormalized using ratio of approximate to exact volumes.



Limitations in DRAGON

20

Storage requirements for tracking file:

- Size of tracking file linear in N :

$$d_t \propto \rho h_+^2 N$$

for $h_+ = 50$ cm, $\rho = 20$ t/cm² and $N = 1000$ regions:

$d_t = 600$ Mb.

- Tracking file limited to disk storage available.
- Use EXCELL: module when disk storage is insufficient for 3-D (2-D not permitted) calculations since tracking file is not saved in this case.



Limitations in DRAGON

21

Storage requirements for CP matrices:

- Number of elements per groups is N^2 .
- Memory space required for execution is about $5N^2$.
- Total disk space required for storage of G group CP is GN^2 .
- For $G = 89$ groups and $N = 1000$ regions: $d_a = 356$ Mb.
- Store ASMPIJ data structure in `XSM_FILE` rather than `LINKED_LIST`.



Limitations in DRAGON

22

Collision probability normalization:

- Collision, leakage and escape probability computed independently.
- Integration procedure preserves symmetry relations.
- Conservation relations generally not satisfied. For example:

$$\sum_{\alpha=1}^{N_{\alpha}} \frac{S_{\alpha}}{4} p_{\alpha j} + \sum_{i=1}^{N_i} \Sigma_i V_i p_{ij} - V_j \neq 0$$

- Conservation relations restored using HELIOS type normalization (default option).

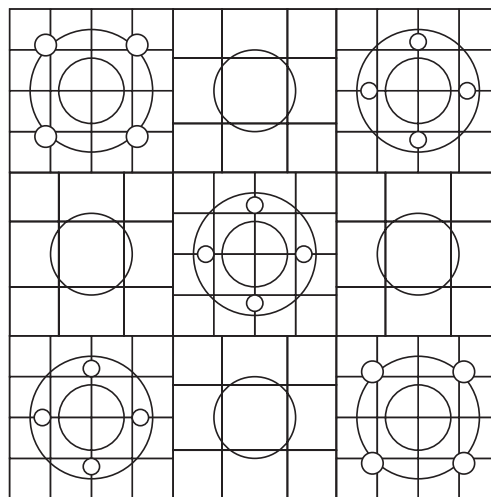


Expectations for DRAGON 3.05

1

Major update to the DRAGON geometries:

- The NXT: tracking module will contain the new EXCELT: tracking option.
- It will be able to analyze more complex geometries in 2-D and 3-D including clusters.



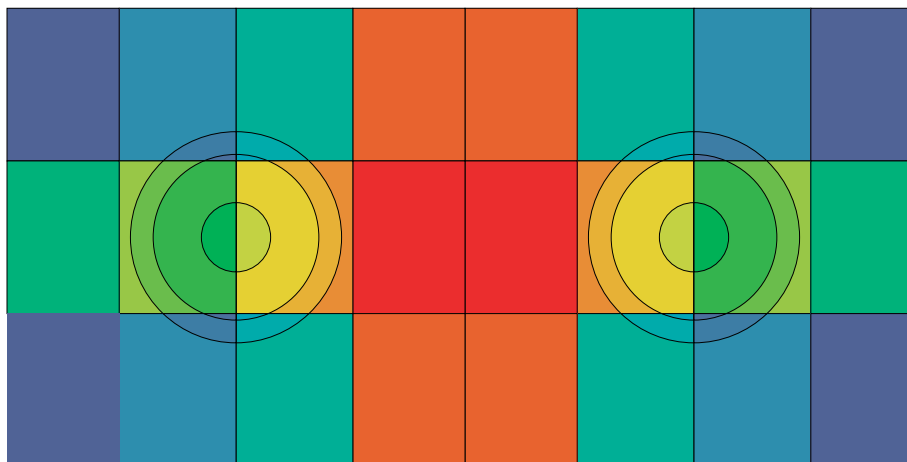


Expectations for DRAGON 3.05

2

What we have now for 2-D Cartesian assemblies:

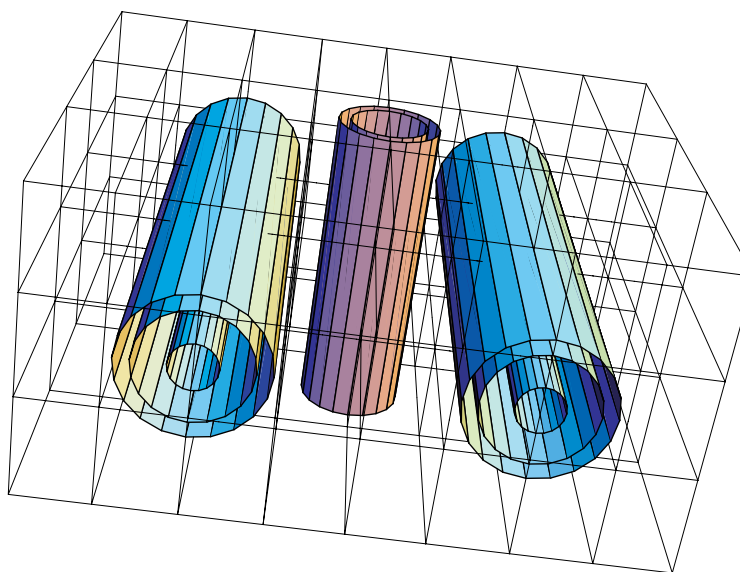
Legend
Color by Region
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18





Expectations for DRAGON 3.05 3

What we have now for 3-D Cartesian assemblies:

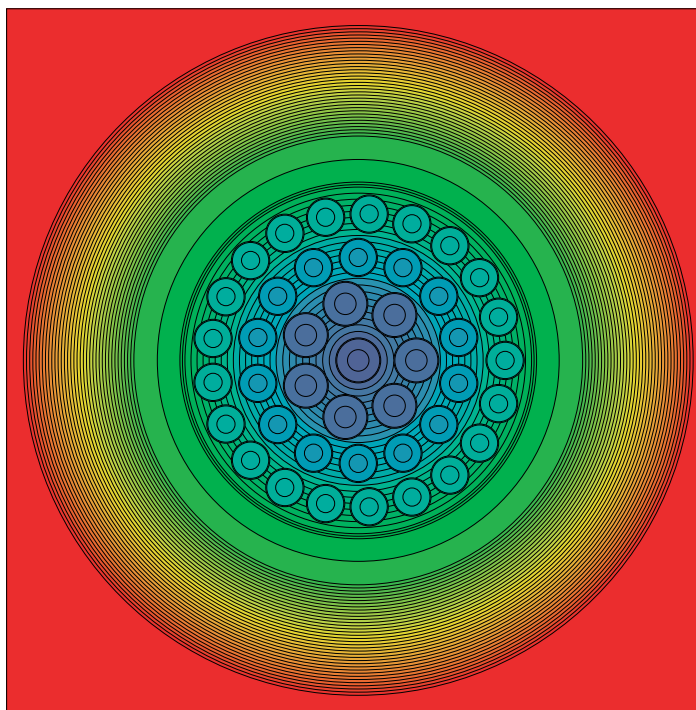




Expectations for DRAGON 3.05

4

What we have now for 2-D cluster cells:

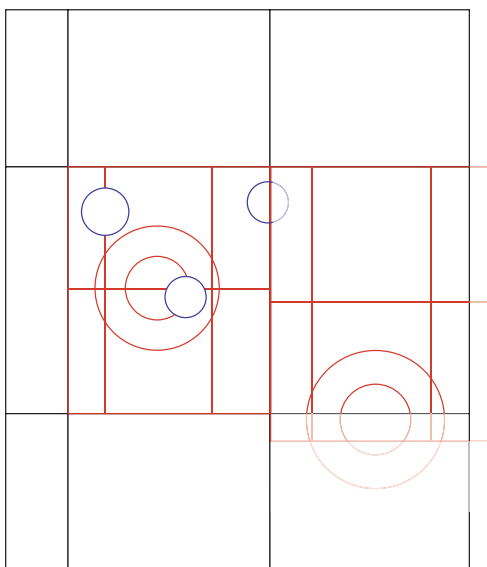




Expectations for DRAGON 3.05

5

What we are in the process of implementing for 2–D assemblies with Cartesian boundaries:



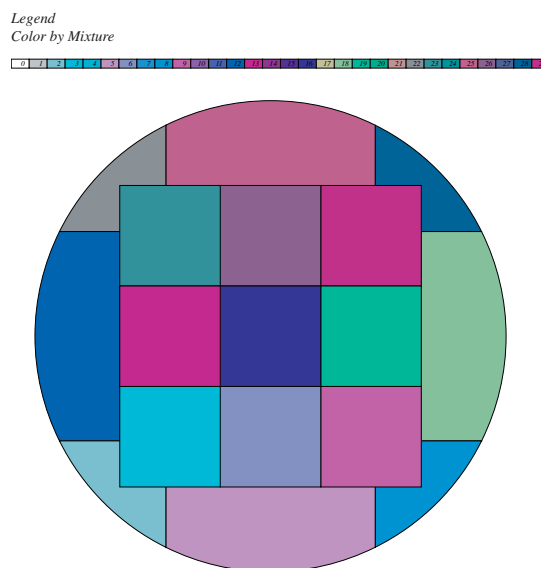
This will also be available in 3–D geometry.



Expectations for DRAGON 3.05

6

What we are in the process of implementing for 2–D assemblies with annular boundaries:



This will also be available in 3–D geometry.

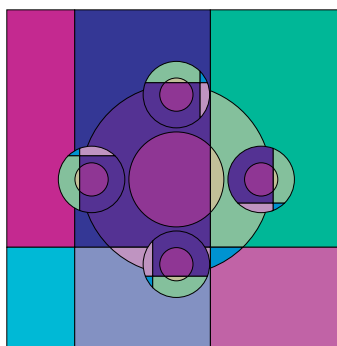


Expectations for DRAGON 3.05

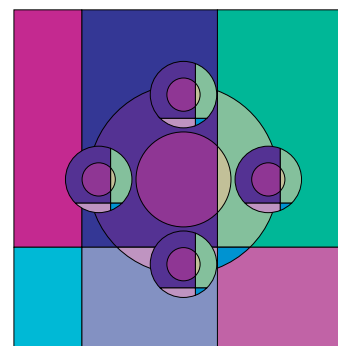
7

What we are in the process of implementing for 2-D assemblies with clusters.

Legend
Color by Mixture



(a) Cluster with pin rotation
(pins located using RPIN)



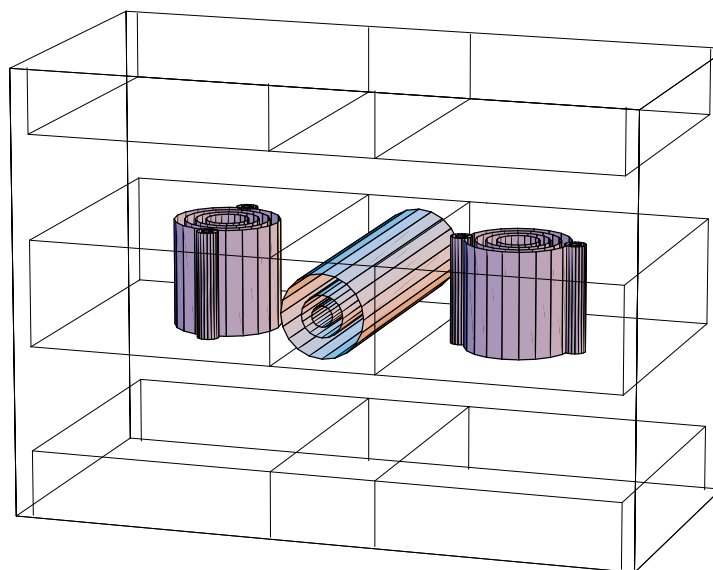
(b) Cluster without pin rotation
(pins located using CPIN)



Expectations for DRAGON 3.05

8

What we are in the process of implementing for 3-D assemblies with clusters.





Towards DRAGON version 4.0

1

1. Microscopic library treatment.
2. Self-shielding options.
3. Method of characteristics.



Towards DRAGON version 4.0

2

Improvements in microscopic cross section library analysis:

- Upgrade of the decay chains treatment to take into account more reactions and to include energy production from other sources than fission (mainly `LIB:` and `EVO:` modules).
- Update the `EDI:` module to process delayed neutron data.
- Upgrade the DRAGLIB format and generate of a DRAGLIB file from ENDF/B-VI using the DRAGR module of NJOY (mainly `LIB:`).
- Upgrade the `LIB:` module in such a way that it can process subgroup data for self-shielding.



Towards DRAGON version 4.0

3

Improvements in the self-shielding options:

- Upgrade the SHI : module to permit distributed self-shielding (Nordheim model).
- Develop the SUBG : module for self-shielding by subgroup.



Towards DRAGON version 4.0

4

Method of characteristics:

- Four versions currently in competition in developmental version of DRAGON:
 1. MOCC method for cyclic tracking (available in 2–D only for mirror like BC)
 2. G. Wu and R. Roy method of characteristics developed for 2–D and 3–D tracking with white BC.
 3. M. Dahmani and R. Roy method of characteristics for anisotropic flux for parallel computers.
 4. R. LeTellier and A. Hébert method of characteristics (available in 2–D only for white BC).
- Rationalize and implement a unique version.